HE
18.5
.A37
no.
DOT-
TSC-
UMTA-
80-36

RT NO. UMTA-MA-06-0048-80-8

MAR 1 1981

MORGANTOWN PEOPLE MOVER
REDUNDANT COMPUTING SYSTEM DESIGN SUMMARY

Jim I. Rucker
Bert J. Hill

BOEING AEROSPACE COMPANY
Automated Transportation Systems
Seattle, Washington 98124

SEPTEMBER 1980

FINAL REPORT

DOCUMENT IS AVAILABLE TO THE PUBLIC
THROUGH THE NATIONAL TECHNICAL
INFORMATION SERVICE, SPRINGFIELD,
VIRGINIA 22161

Prepared for

U.S. DEPARTMENT OF TRANSPORTATION
URBAN MASS TRANSPORTATION ADMINISTRATION
Office of Technology Development and Deployment
Office of AGT Applications
Washington, DC 20590

NOTICE

This document is disseminated under the sponsorship of the Department of Transportation in the interest of information exchange. The United States Government assumes no liability for its contents or use thereof.

NOTICE

The United States Government does not endorse products or manufacturers. Trade or manufacturers' names appear herein solely because they are considered essential to the object of this report.

, A37
no.
DOT-
TSC
UMTA-
80-36

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
UMTA-MA-06-0048-80-8	PB 81-151367	

4. Title and Subtitle	5. Report Date
MORGANTOWN PEOPLE MOVER REDUNDANT COMPUTING SYSTEM DESIGN SUMMARY	September 1980
	6. Performing Organization Code DTS-723
7. Author(s) Jim I. Rucker Bert J. Hill	8. Performing Organization Report No. DOT-TSC-UMTA-80-36
9. Performing Organization Name and Address Boeing Aerospace Company* Automated Transportation Systems Seattle, WA 98124	10. Work Unit No. (TRAIS) UM041/R0725
	11. Contract or Grant No. DOT-TSC
12. Sponsoring Agency Name and Address U.S. Department of Transportation Urban Mass Transportation Administration Office of Technology Development and Deployment Washington, DC 20590	13. Type of Report and Period Covered Final Report
	14. Sponsoring Agency Code UTD-60

15. Supplementary Notes

*Under contract to: U.S. Department of Transportation
Research and Special Programs Administration
Transportation Systems Center, Kendall Sq.,
Cambridge, MA 02142

16. Abstract

This report describes the redundant computing design used for the current (1980 Phase II) Morgantown People Mover (MPM) system. Since the MPM system was developed in three phases, this report presents some historical data leading to the current design. This report also includes results of experience with the redundant computing system, plans for potential system improvement, and recommendations so that future system designers can benefit from the experience gained in developing the Phase II MPM System.

17. Key Words	18. Distribution Statement
- Redundant, Prime String, Backup String, Reconfiguration, Syncronization, Morgantown Software, AGT	DOCUMENT IS AVAILABLE TO THE PUBLIC THROUGH THE NATIONAL TECHNICAL INFORMATION SERVICE, SPRINGFIELD, VIRGINIA 22161

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
UNCLASSIFIED	UNCLASSIFIED	158	

Preface

This report describes the redundant computing design used for the current (1980 Phase II) Morgantown People Mover (MPM) system. Since the MPM system was developed in three phases, this report presents some historical data leading to the current design. This report also includes results of experience with the redundant computing system, plans for potential system improvement, and recommendations so that future system designers can benefit from the experience gained in developing the Phase II MPM System.

This work described in this report was sponsored by the Office of AGT Applications, Office of Technology Development and Deloyment of the U.S. Department of Transportation's Urban Mass Transportation Administration. This report was monitored by U.S. Department of Transportation, Transportation Systems Center (TSC), Cambridge, Massachusetts.

The bulk of the design of the redundant computing system was accomplished by Boeing Aersopace Company, Seattle, Washington. Some of the early trade studies were performed by Jet Propulsion Laboratory (JPL), Pasadena, California.

METRIC CONVERSION FACTORS

Approximate Conversions to Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
in	inches	2.5	centimeters	cm
ft	feet	30	centimeters	cm
yd	yards	0.9	meters	m
mi	miles	1.6	kilometers	km
		AREA		
in^2	square inches	6.5	square centimeters	cm^2
ft^2	square feet	0.09	square meters	m^2
yd^2	square yards	0.8	square meters	m^2
mi^2	square miles	2.6	square kilometers	km^2
	acres	0.4	hectares	ha
		MASS (weight)		
oz	ounces	28	grams	g
lb	pounds	0.45	kilograms	kg
	short tons (2000 lb)	0.9	tonnes	t
		VOLUME		
tsp	teaspoons	5	milliliters	ml
Tbsp	tablespoons	15	milliliters	ml
fl oz	fluid ounces	30	milliliters	ml
c	cups	0.24	liters	l
pt	pints	0.47	liters	l
qt	quarts	0.95	liters	l
gal	gallons	3.8	liters	l
ft^3	cubic feet	0.03	cubic meters	m^3
yd^3	cubic yards	0.76	cubic meters	m^3
		TEMPERATURE (exact)		
°F	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C

Approximate Conversions from Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
		LENGTH		
mm	millimeters	0.04	inches	in
cm	centimeters	0.4	inches	in
m	meters	3.3	feet	ft
m	meters	1.1	yards	yd
km	kilometers	0.6	miles	mi
		AREA		
cm^2	square centimeters	0.16	square inches	in^2
m^2	square meters	1.2	square yards	yd^2
km^2	square kilometers	0.4	square miles	mi^2
ha	hectares (10,000 m^2)	2.5	acres	
		MASS (weight)		
g	grams	0.035	ounces	oz
kg	kilograms	2.2	pounds	lb
t	tonnes (1000 kg)	1.1	short tons	
		VOLUME		
ml	milliliters	0.03	fluid ounces	fl oz
l	liters	2.1	pints	pt
l	liters	1.06	quarts	qt
l	liters	0.26	gallons	gal
m^3	cubic meters	35	cubic feet	ft^3
m^3	cubic meters	1.3	cubic yards	yd^3
		TEMPERATURE (exact)		
°C	Celsius temperature	9/5 (then add 32)	Fahrenheit temperature	°F

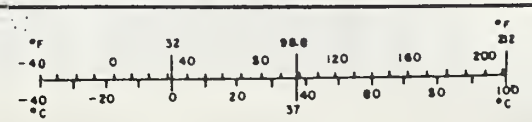


TABLE OF CONTENTS

Section			Page
1.	INTRODUCTION		1
	1.1 GENERAL		
	1.2 MPM SYSTEM HISTORY		1
	1.3 REDUNDANT COMPUTING SYSTEM HISTORY		4
2.	SYSTEM DESCRIPTION		12
	2.1 MPM SYSTEM DESCRIPTION		12
	2.2 COMPUTER HARDWARE SYSTEM DESCRIPTION		18
	2.3 COMPUTER SOFTWARE SYSTEM DESCRIPTION		24
	2.4 HARDWARE/SOFTWARE FUNCTIONAL ALLOCATION		36
	2.4.1	Functions Performed by Hardware in Redundant Elements	37
	2.4.2	Functions Performed by Software in Redundant Elements	38
3.	REQUIREMENTS AND DESIGN		40
	3.1 REQUIREMENTS FOR THE DESIGN OF REDUNDANT COMPUTING SYSTEM ELEMENTS		40
	3.1.1	General Requirements	41
	3.1.2	Hardware Requirements	43
	3.1.3	Software Requirements	49
	3.2 DETAILED DESIGN OF THE REDUNDANT COMPUTING SYSTEM HARDWARE ELEMENTS		51
	3.2.1	Data Acquisition Unit (DAU) Input SPE Equipment	51
	3.2.2	Destination Selection Unit and Vehicle Downlink Input SPE Equipment	60
	3.2.3	Vehicle Uplink, Collision Avoidance, and Mimic Output SPE Equipment	60
	3.2.4	Passenger Boarding Displays and Electrification Trip SPE Equipment	61
	3.2.5	Modem Reconfiguration Unit	61

TABLE OF CONTENTS (Continued)

Section			Page
	3.3	DETAILED DESIGN OF THE REDUNDANT COMPUTING SYSTEM SOFTWARE ELEMENTS	63
		3.3.1 Executive Service Requests	65
		3.3.2 I/O Management and Interrupt Handling	74
		3.3.3 Data Synchronization	81
		3.3.4 System Status Monitoring	85
		3.3.5 Configuration Control	96
		3.3.6 Reconfiguration Control	97
	3.4	OFF-THE-SHELF VS. SPECIAL DESIGN COMPONENTS	106
		3.4.1 Hardware Components	109
		3.4.2 Software Components	109
4.	ANALYSIS AND TEST RESULTS		111
	4.1	MPM REDUNDANT COMPUTING SYSTEM OPERATING RESULTS	111
	4.2	ASSESSMENT OF MPM REDUNDANT COMPUTING SYSTEM	119
	4.3	PROBLEMS ENCOUNTERED AND SOLUTIONS IMPLEMENTED	128
5.	POTENTIAL SYSTEM IMPROVEMENTS		132
6.	RECOMMENDATIONS		139
	APPENDIX A - GLOSSARY		A1
	APPENDIX B - REPORT OF NEW TECHNOLOGY		B1

LIST OF ILLUSTRATIONS

Figure		Page
1-1	Program History	2
1-2	MPM System Elements (Phases IA and IB)	3
1-3	MPM System Elements (Phase II)	5
1-4	MPM Redundant Computing System History	5
1-5	Phase IA Computing System	7
1-6	Phase IB Redundant Computing System	8
1-7	Phase II Redundant Computing System	11
2-1	MPM Vehicle Functional Schematic	14
2-2	Guideway System	15
2-3	C&CS Configuration	17
2-4	C&CS Functional Diagram	17
2-5	MPM Phase II Redundant Computing System	19
2-6	MPM Central Computer Configuration	20
2-7	MPM Typical Station Computer Configuration	22
2-8	MPM Operational Software Subsystem Organization	25
2-9	Operational Software Organizational Levels	26
2-10	Central Applications Program Organization	27
2-11	Passenger Station Applications Program Organization	30
2-12	Maintenance Station Applications Program Organization	32
2-13	Executive Program Organization	34
3-1	Interface Between Station Computers and Station Electronics Data Acquisition Unit (DAU)	45
3-2	DAU Data Interface Timing Requirements	46
3-3	Station SPE State Transition Requirements	48
3-4	DAU SPE Functional Block Diagram	52
3-5	DAU Data Single String Timing	53
3-6	DAU Data Dual String Timing (A String Prime)	54
3-7	Vehicle Uplink SPE Functional Block Diagram	61
3-8	Modem Reconfiguration Unit Functional Diagram	62
3-9	Executive Routines which Support Redundancy within Executive Program Organization	64
3-10	Arm Backup Computer System ESR	66
3-11	Arm Backup Control/Data Flow	67
3-12	System Status ESR	68
3-13	System Status Control/Data Flow	69
3-14	Switchover ESR	70
3-15	Switchover ESR Control/Data Flow	71
3-16	Activate SPE ESR	73
3-17	Activate SPE ESR Control/Data Flow	74
3-18	DR11A/DR11C Input Done (DSU, DHU, DAU)	76
3-19	DR11A/DR11C Input Done Control/Data Flow	77
3-20	CRT Input Interrupt Service	78
3-21	Bus Link Interrupt Service	79

Figure		Page
3-22	CRT Input Interrupt Service and Bus Link Interrupt Service Control/Data Flow	80
3-23	Data Synchronization	83
3-24	Data Synchronization Timeout	86
3-25	Data Synchronization Control/Data Flow	87
3-26	Worst Case Other-Central Failure Detection Time Line	88
3-27	Other-Central Monitoring	90
3-28	Other-Central Monitoring Control/Data Flow	92
3-29	Special Purpose Equipment (SPE) Monitoring	93
3-30	Special Purpose Equipment (SPE) Monitoring Control/Data Flow	95
3-31	Central Loader	97
3-32	Central Loader Control/Data Flow	98
3-33	Central Reconfiguration Decision	100
3-34	Central Reconfiguration Processing	103
3-35	Central Reconfiguration Decision and Processing Control/Data Flow	104
3-36	Station Reconfiguration Decision	105
3-37	Station Reconfiguration Processing	107
3-38	Station Reconfiguration Decision and Processing Control/Data Flow	108
4-1	Measurement of Switchover Time	118
4-2	Switchover Time Line - Breakdown of Measured Time	120
4-3	Switchover Time Line - Worst Case Analysis	121
4-4	MPM Redundant String Auto Boot Panel	125
5-1	Current Central Average CPU Utilization	133
5-2	Projected Central Average CPU Utilization with Improved Communication Interfaces	133
5-3	Backup Central Eavesdrop on Prime to Compensate for Failed Station in Backup String	135
5-4	Adjacent Sets of Redundant Computing Strings	137
5-5	Regions With a Supervisory Central	138

Table		Page
3-1	DAU Data Single String Timing	53
3-2	DAU Data Dual String Timing	54
4-1	MPM Computer Hardware and Software Availability - Required/Estimated Versus Measured Comparison	112

1. INTRODUCTION

1.1 General

The purpose of this report is to describe the redundant computing system design used for the current (1980 Phase II) Morgantown People Mover (MPM) system. Since MPM was developed in three phases, the third phase currently in public service, this report presents some historical data leading to the current design. Some reliability data resulting from field measurements, an assessment of the redundant computing system, and recommendations important for future uses of this redundant computing system approach are presented. This report consists of the following sections:

Section
1. Introduction
2. System Description
3. Requirements and Design
4. Analysis and Test Results
5. Potential System Improvements
6. Recommendations

The redundant computing system is that part of the Control and Communications System (C&CS) consisting of redundant computer hardware and software and the special purpose equipment (SPE) used to interface the dual computing system to the rest of the C&CS system.

1.2 MPM System History

Morgantown, West Virginia is a city of 29,500 population with a long standing transportation problem. The city has an inadequate street network, imposed by difficult topographical constraints and complicated by main arteries which are steep and narrow. There are no major freeways. West Virginia University (23,000 students and staff) with its four

separated campuses 1.5 miles apart, competes with the city traffic for the same transportation facilities. Because of the congestion produced by heavy flow on the steep and narrow roads, average speeds in peak traffic periods are reduced to five to ten miles per hour. These conditions plus extremely varied weather conditions made Morgantown an excellent site for demonstration of a new transportation system technology.

The Morgantown project, which began in 1969, is an Urban Mass Transportation Administration (UMTA) demonstration program that provides a personal rapid transit system (Figure 1-2) between the central business district of Morgantown, West Virginia, and the widely separated campuses of West Virginia University (WVU). The MPM system is an automated, two-mode (schedule and demand) transit system that consists of a fleet of electrically powered, rubber-tired, passenger-carrying vehicles operating on a dedicated guideway network under the redundant computing system computer control.

Figure 1-1 shows the MPM system program history.

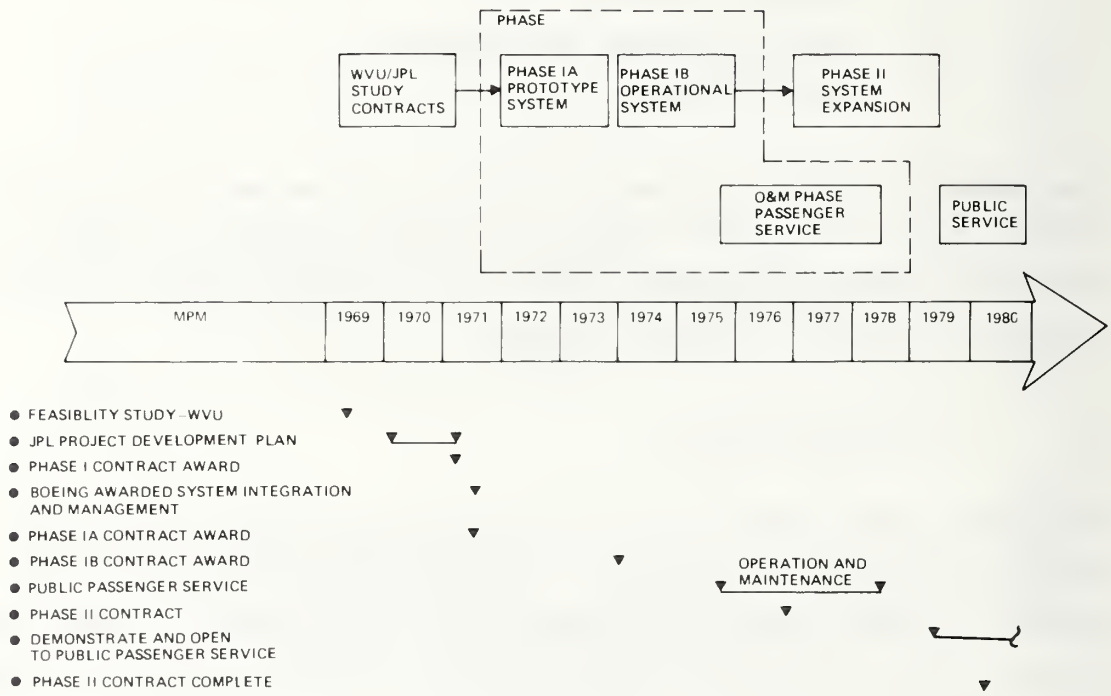


FIGURE 1-1. PROGRAM HISTORY

The project began with a research grant given to WVU in 1969. Initially, it was to be an expanded version of a system already developed by the Alden Company of Natick, Massachusetts. However, in mid-1970 it was determined that a new system would be created under requirements and constraints established jointly between WVU and UMTA. The Jet Propulsion Laboratory (JPL) was selected as system manager and designer in 1970. JPL performed the preliminary analyses and trade studies. Contracts were let to Boeing for vehicle design and fabrication in May 1971 and to Bendix Company for communications and control of a six-station system. In September 1971, with much of the system design completed, UMTA transferred system management responsibility from JPL to The Boeing Company. Also, at this time, the program was phased first to build a Phase I three-station system (Figure 1-2) and later in Phase II to expand to a six-station system in Phase II.

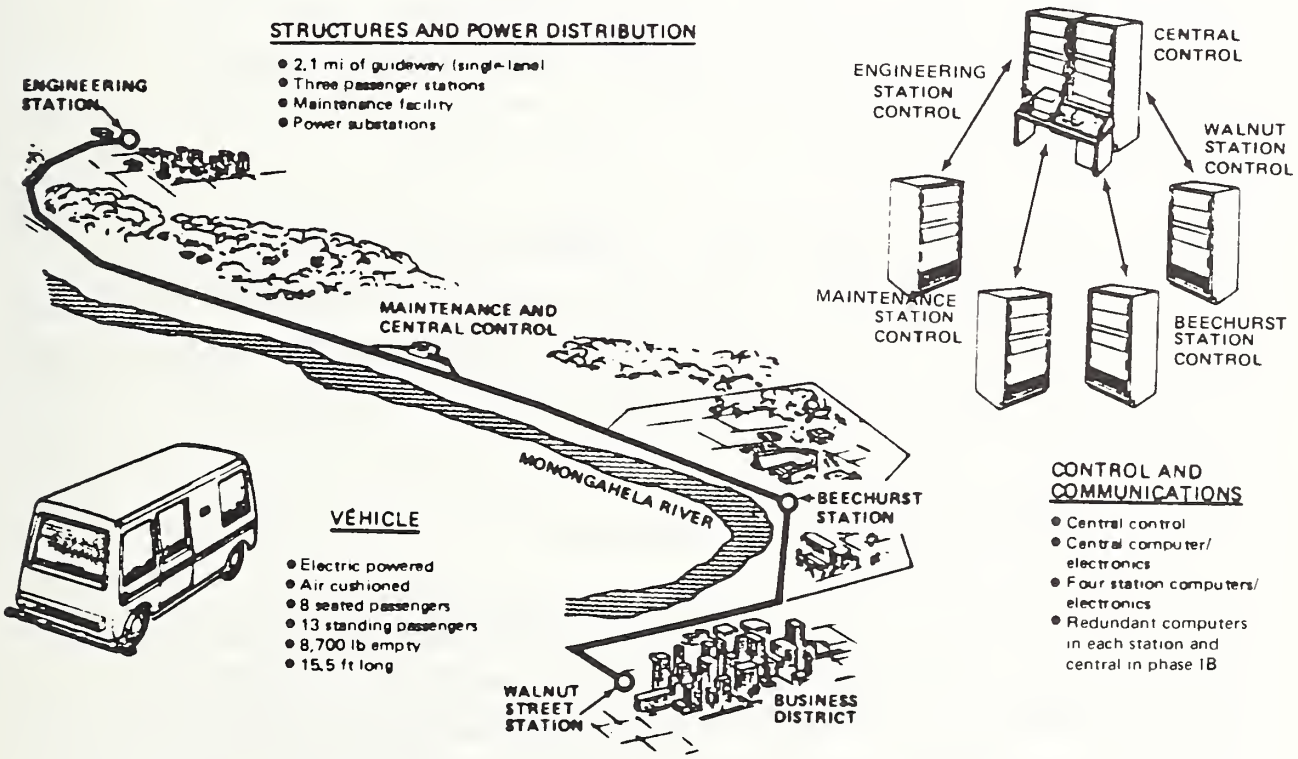


FIGURE 1-2. MPM SYSTEM ELEMENTS (PHASES IA AND IB)

Phase I was divided into a Phase IA and a Phase IB. Phase IA, completed in September 1973, resulted in a prototype system comprising 5.2 miles of single lane guideway, three passenger stations, a maintenance and central control facility, and five test vehicles. Phase IB provided the additional facilities required for public service including a fleet of 45 vehicles. Phase IB also provided the opportunity to resolve the problems encountered in the prototype Phase IA system. Phase IB testing concluded with the system being opened to passenger service in September 1975. The system is of modular design and allowed growth from the Phase IB configuration to an expanded configuration which accommodates 73 vehicles and two new stations for the Phase II expansion. In November 1976 approval of the Phase II MPM program was granted. The Phase IB system was removed from passenger service in July 1978 to allow Phase II construction. Design and construction of two new stations, 3.4 miles of single-lane guideway, and 28 new vehicles, along with certain rework and retrofit tasks were completed in mid 1979. A system demonstration period with public passenger service took place between July 1979 and March 1980. After this demonstration the system will be continue to be in revenue service.

Figure 1-3 shows the present Phase II guideway configuration and the three basic system elements: the vehicle system, the structures and power distribution system, and the control and communications system (C&CS). The redundant computing system portion of the C&CS is the subject of this report.

1.3 Redundant Computing System History

Figure 1-4 shows a top level history of the MPM computing system. JPL conducted a centralized vs. distributed trade study in April 1971. A purely centralized architecture was rejected because it would have placed a heavy burden on the central computer hardware/software to accomplish the timing required to handle the entire system data rate. A purely distributed system with a computer at each station was rejected

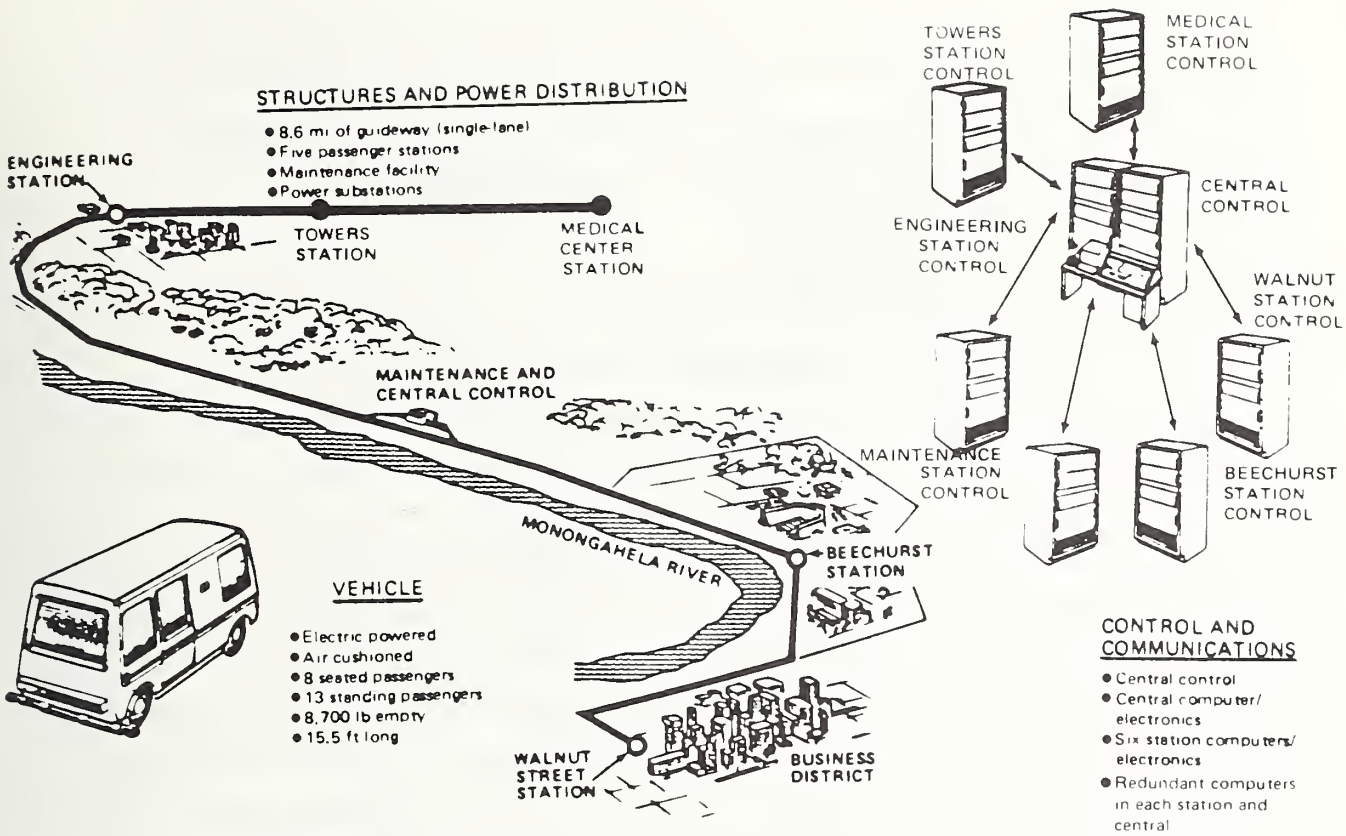


FIGURE 1-3. MPM SYSTEM ELEMENTS (PHASE II)

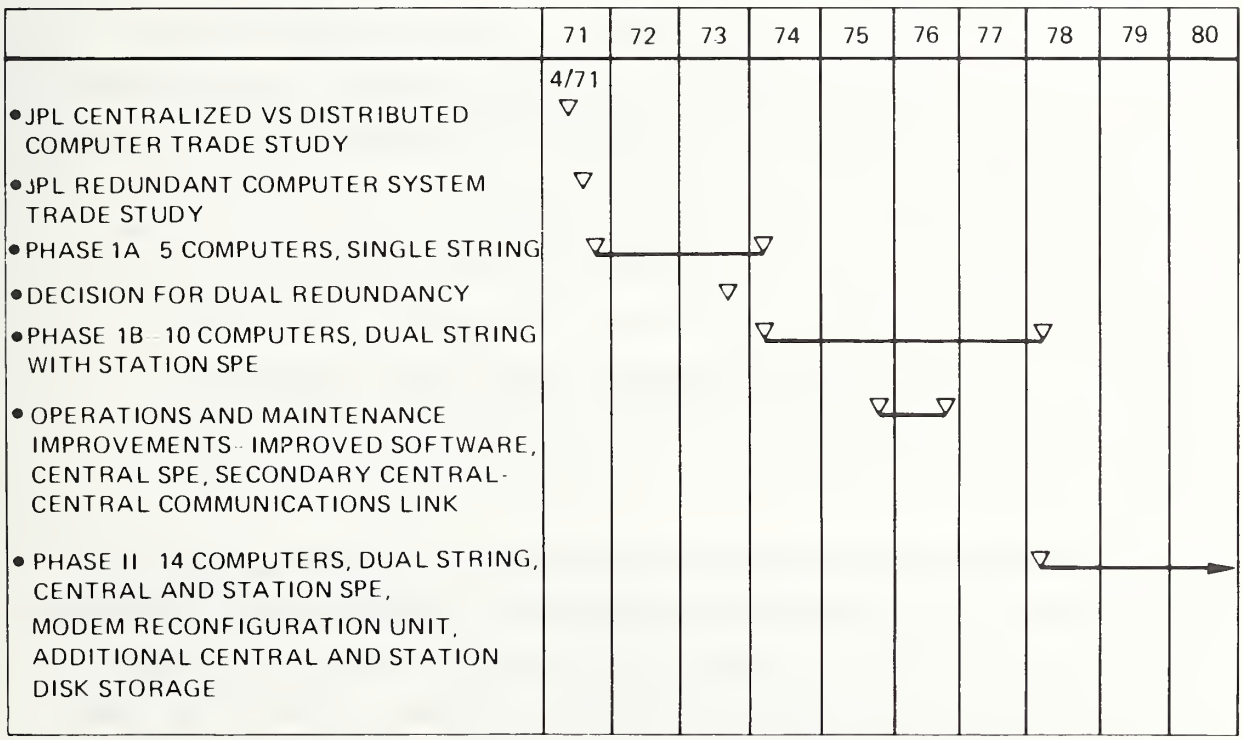

71
72
73
74
75
76
77
78
79
80
4/71
• JPL CENTRALIZED VS DISTRIBUTED COMPUTER TRADE STUDY
• JPL REDUNDANT COMPUTER SYSTEM TRADE STUDY
• PHASE 1A 5 COMPUTERS, SINGLE STRING
• DECISION FOR DUAL REDUNDANCY
• PHASE 1B 10 COMPUTERS, DUAL STRING WITH STATION SPE
• OPERATIONS AND MAINTENANCE IMPROVEMENTS IMPROVED SOFTWARE, CENTRAL SPE, SECONDARY CENTRAL-CENTRAL COMMUNICATIONS LINK
• PHASE II 14 COMPUTERS, DUAL STRING, CENTRAL AND STATION SPE, MODEM RECONFIGURATION UNIT, ADDITIONAL CENTRAL AND STATION DISK STORAGE

FIGURE 1-4. MPM REDUNDANT COMPUTING SYSTEM HISTORY

as not suitable since each station would have had to duplicate core resident status of the entire guideway to accomplish the vehicle dispatch function and because the burden on the computers required to handover vehicles from station to station would have been heavy.

Examination of the requirements showed that all functions, except those for control of the local station, should be performed at a central location. This led to a trade between a centralized approach with a large central computer and special purpose station hardware and a distributed approach with a central computer and small general purpose computers located at each station. The trade study showed that the two approaches were nearly equal in cost, reliability, maintainability, and safety, but the distributed approach was chosen because its modular architecture had more inherent growth capability. In mid 1971, JPL studies showed that some type of redundant computing system was required. This was apparent because of the major contribution to system operation that the computing system makes and because of the requirement that to the greatest extent possible no single failure may cause system downtime or degrade system operation.

When the system management responsibility was transferred from JPL to Boeing the program was phased for budgetary reasons first to build a Phase IA four station single computing string system with the redundant computing system design and implementation deferred until Phase IB. The Phase IA computing system shown in Figure 1-5 consisted of a DEC PDP 11/20 computer at each of four stations (the Maintenance facility is considered a "station") and a PDP 11/20 computer at the central facility. The DEC PDP 11 minicomputers were chosen for MPM based on their ability to satisfy the requirements existing at the time and on their expansion flexibility to provide for subsequent growth.

The field operating experience of the Phase IA single string prototype system made it very obvious that some sort of redundant computing system would be required for the Phase IB system since the mean time between failure (MTBF) for the phase 1A computer system was 136 hours and since the repair times were long.

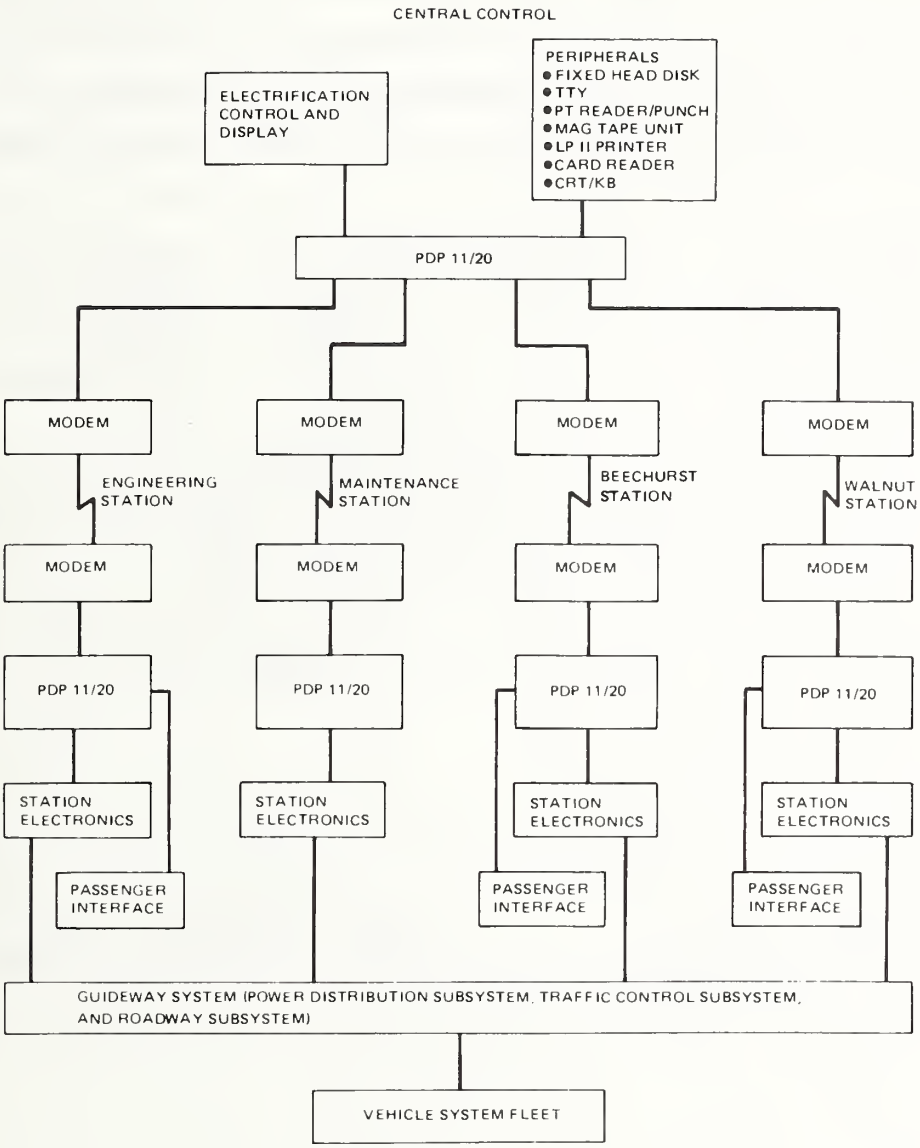


FIGURE 1-5. PHASE IA COMPUTING SYSTEM

For Phase IB the overall system availability target was 0.9630 with a system MTBF of twenty-six hours and mean downtime of one hour. Availability is defined as the probability that the system is ready for use at any random point in time. Of this system target the computing system was allocated an availability of 0.9946 with a MTBF of 500 hours and a mean downtime of 2.7 hours.

Thus, the least costly computer system which would meet this availability requirement was sought, and the Phase IB redundant computing system shown in Figure 1-6 was the result. The computing system was made dual by designing special purpose equipment (SPE) to interface the dual station computers to the existing single string station hardware.

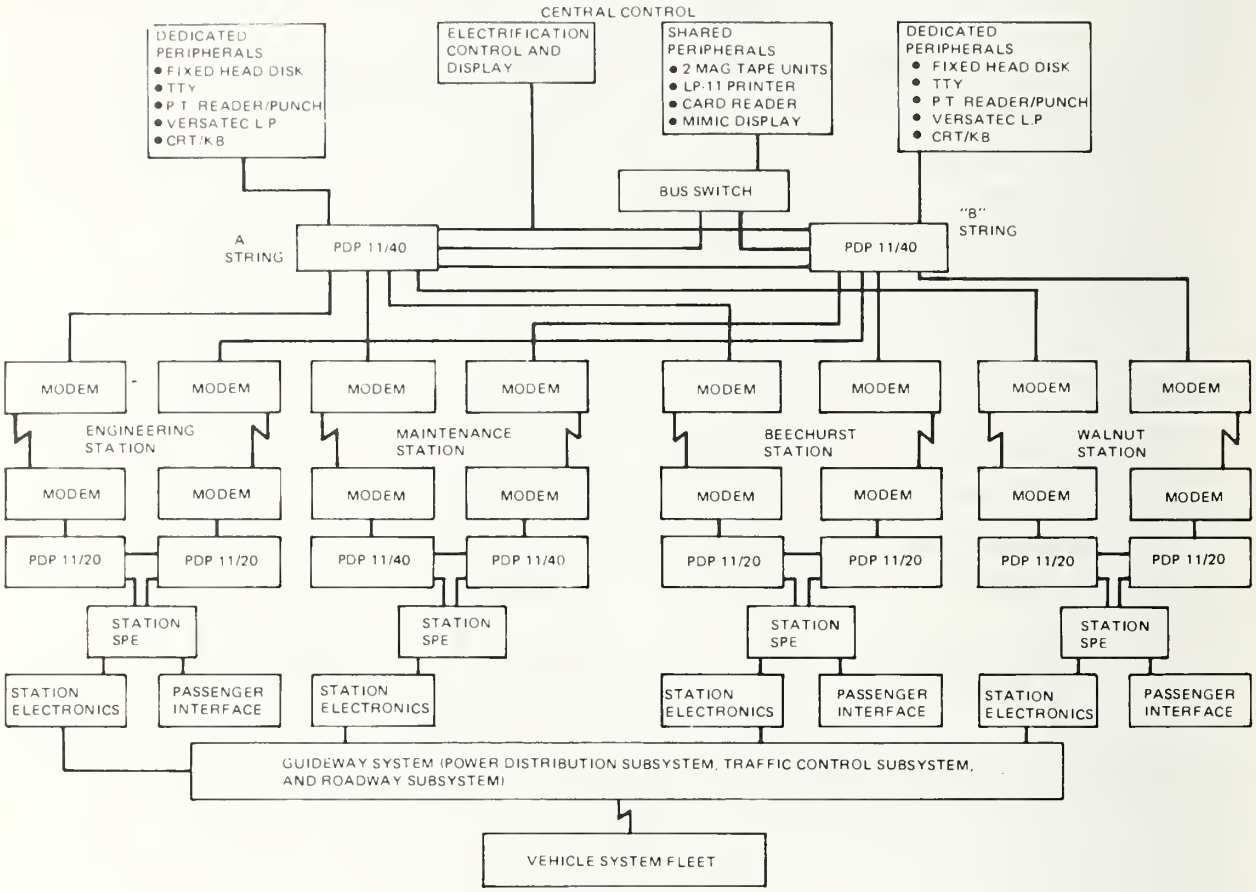


FIGURE 1-6. PHASE IB REDUNDANT COMPUTING SYSTEM

The Phase IB operational software was developed by Boeing and System Development Corporation (SDC) of Santa Monica, California. SDC designed, coded, and module tested 80% of the application software while Boeing developed the rest of the application software, developed the executive software, and integrated and verified the entire operational software package. A Boeing built executive was chosen over an off-the-shelf executive for the following reasons: it provided all the required capabilities; it provided a simpler executive to application software

interface making the subcontracted application software easier to integrate and verify; it was much more efficient since it was tailored to the MPM system configuration; it was considered less costly to extend the executive for Phase II; and it provided a common executive for the central and station allowing software development compatibility between the central and station software. Off-the-shelf executives considered would have dictated different executives and hence different control concepts for the central and station software. A disadvantage of the Boeing built executive was the cost to develop and verify the executive. However, the simplicity of the executive made the subcontracted applications software easier to integrate and verify; in addition, part of the executive development cost was paid with company research and development funds. If a similar trade study were performed today, an off-the-shelf executive would probably be chosen since off-the-shelf executives available today are more suited to real-time control systems than they were in 1973. Commercial distributed networks are also much more common today than they were then.

During a one year operations and maintenance contract at the end of Phase IB the operational software was made more reliable and capable, and a central SPE was added to the central hardware so that the central mimic board which dynamically shows the vehicle location throughout the system could be automatically switched by software from the failed to the new prime controlling computing string. Also, a second communications link was added between the two central computers to allow reconfiguration arbitration in the event of failure of the central-central communications link.

During the last year of operation of the Phase IB system (July 1977 through July 1978) the overall system availability was 0.9772. Experience from the Phase IB system showed that the MTBF of 500 hours for the computing system was unrealistic since the individual computers failed more often than estimated. However, the availability of the computing system was close to the requirement because the mean downtime was smaller than anticipated.

For the Phase II system the system availability target was 0.9700 with a MTBF of 9.2 hours and a mean downtime of 0.28 hours. The computing system was allocated an availability of 0.9969 with a MTBF of 175 hours and a mean downtime of 0.54 hours. In order for the larger Phase II computing system to meet the increased availability requirements, reliability improvements were needed in Phase II.

The Phase II redundant computing system is shown in Figure 1-7. For Phase II, two new passenger stations were added with dual PDP 11/40 computers, existing stations were upgraded from PDP 11/20's to PDP 11/40's to provide additional memory capacity, and central was upgraded from PDP 11/40's to PDP 11/55's to provide increased speed to handle the additional comunications required by the two new stations. Reliability improvements for Phase II included a modem reconfiguration unit, cartridge disks at central, and floppy disks at the stations. The modem reconfiguration unit allows station computers to be manually switched between A and B strings. The cartridge disks allow vehicle status files to be moved between strings to recover from computer component failures and the floppy disks allow the station software image to be recorded at the station providing rapid system reload and restart. These reliability changes increase the availability primarily by reducing the mean downtime. In addition, the upgrading to newer computers increases the reliability and makes failures easier to isolate.

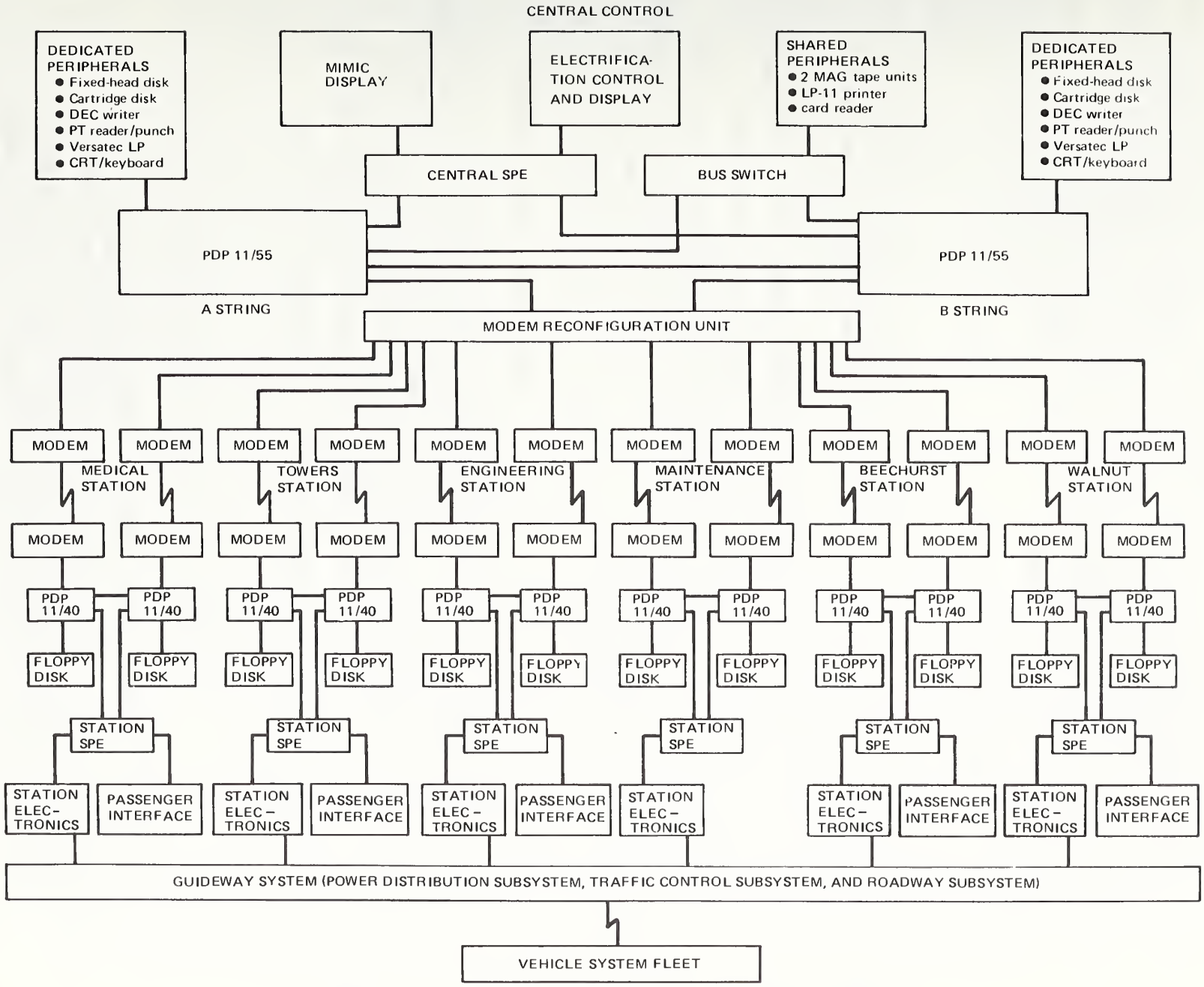


FIGURE 1-7. PHASE II REDUNDANT COMPUTING SYSTEM

2. SYSTEM DESCRIPTION

This section includes a brief description of the MPM system with emphasis on the redundant elements of the redundant computing system. As explained previously, MPM was developed in phases. However, the baseline system to be described in this report is the Phase II system which entered passenger service in July 1979.

2.1 MPM System Description

MPM is an automated guideway transit system which provides personal rapid transit service between the separated campuses of West Virginia University and the Morgantown central business district. The system consists of a fleet of 73 electrically powered, rubber tired, passenger carrying vehicles operating up to 30 miles per hour on a dedicated guideway network at 15 second headways (vehicle separation). The system provides a safe, comfortable, low polluting, reliable means of transportation. The system features year round operation, as well as direct nonstop origin to destination service.

The MPM is operated in either schedule or demand mode. During those periods when passenger demand is highly predictable, the system is operated in schedule mode. Vehicles are dispatched between origin/destination pairs on a preset schedule. When passenger demand is less predictable, the system is operated in demand mode. Vehicles are then dispatched only in response to a passenger request. Passenger actions upon entering the system are always the same regardless of the mode in which the system is operating.

Operation of the MPM system, as summarized from a passenger's viewpoint, is as follows: arrive on concourse level of origin station where static and dynamic displays provide direction to the platform servicing his destination; proceed to the platform level; insert a coded card or exact change in a fare gate destination selection unit and press a button selecting destination. A gate display illuminates informing

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
UMTA-MA-06-0048-80-8	PB 81-151367	

4. Title and Subtitle	5. Report Date
Morgantown People Mover Redundant Computing System Design Summary.	September 1980
	6. Performing Organization Code: DTS-723
7. Author(s): Jim I. Rucker and Bert J. Hill	8. Performing Organization Report No.: DOT-TSC-UMTA-80-36
9. Performing Organization Name and Address: Boeing Aerospace Company* Automated Transportation Systems Seattle, Washington 98124	10. Work Unit No. (TRAIS): MA-06-0048(UM041/R0725)
	11. Contract or Grant No.: MA-06-0048
12. Sponsoring Agency Name and Address: U.S. Department of Transportation Urban Mass Transportation Administration 400 Seventh Street, S.W. Washington, DC 20590	13. Type of Report and Period Covered: Final Report
	14. Sponsoring Agency Code: UTD-60

15. Supplementary Notes

*under contract to: U.S. Department of Transportation
Research and Special Programs Administration
Transportation Systems Center
Cambridge, Massachusetts 02142

16. Abstract

The purpose of this report is to describe the redundant computing system design used for the current 1980 Phase II Morgantown People Mover (MPM) system. The redundant computing system is that part of the control and communications system (C&CS) consisting of redundant computer hardware and software and the special purpose equipment (SPE) used to interface the dual computing system to the rest of the C&CS system.

The Morgantown project, which began in 1969, is an Urban Mass Transportation Administration program that provides a personal rapid transit system between the central business district of Morgantown, West Virginia, and the widely separated campuses of West Virginia University. The MPM system is an automated, two-mode (schedule and demand) transit system that consists of a fleet of electrically powered, rubber-tired, passenger-carrying vehicles operating on a dedicated guideway network under the redundant computing system computer control.

Since the MPM system was developed in three phases, this report presents some historical data leading to the current design. The report also includes results of experience with the redundant computing system, plans for potential system improvement, and recommendations so that future system designers can benefit from the experience gained in developing the Phase II MPM system.

17. Key Words	18. Distribution Statement
AGT; Automated Guideway Transit; Control and Communications Systems; Morgantown People Mover; MPM; Personal Rapid Transit; PRT; Redundant Computing System; Software; Morgantown, West Virginia	Available to the public through the National Technical Information Service, Springfield, Virginia 22161.

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified		A08

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

passenger to "proceed" to the vehicle loading area. A vehicle destination passenger boarding display above the loading gate provides vehicle boarding instructions. If assistance is needed for any reason, a dedicated telephone link to the central operator is available near each entry gate area. The passenger is kept informed of changes in the system operating status via station public address system. The passenger boards a vehicle when it arrives at the loading gate, and the display indicates the vehicle is assigned to his destination. The door closes and the vehicle automatically proceeds non-stop to his destination. At the destination station the vehicle stops at an unloading gate, the door opens and the passenger leaves the station through an exit gate. Elevator service is provided from station concourse levels to each platform to permit use of the system by the handicapped and elderly.

The MPM system is comprised of three major system elements: the vehicle system, the structures and power distribution system, and the control and communications system (C&CS).

Vehicle System. The vehicle system is illustrated in Figure 2-1. The vehicles are relatively small, carrying up to 21 passengers - 8 seated and 13 standing. The vehicle size has been selected to provide economical service during both peak and low demand periods. The vehicle length is 15.5 feet and its width is 6 feet. Its weight is approximately 8600 pounds empty. Speeds up to 30 mph are provided with DC electric motor propulsion. Rubber tires and an air spring suspension provide a quiet and comfortable ride. The vehicle responds to remote controls and commands from the computerized control and communications system to provide completely automatic operation. Steering guide wheels follow rails mounted on the side of the guideway. Either the left or right steering guide wheel is used depending on the desired route.

Structure and Power Distribution System. The Structure and Power Distribution System (S&PDS) includes 8.6 lane miles of guideway, the station facilities, the central control and communications facility, the maintenance facility, and the electrical power substations and distribution system. The guideway system is illustrated in Figure 2-2. The main guideway is a limited access route connecting the five passenger stations. The

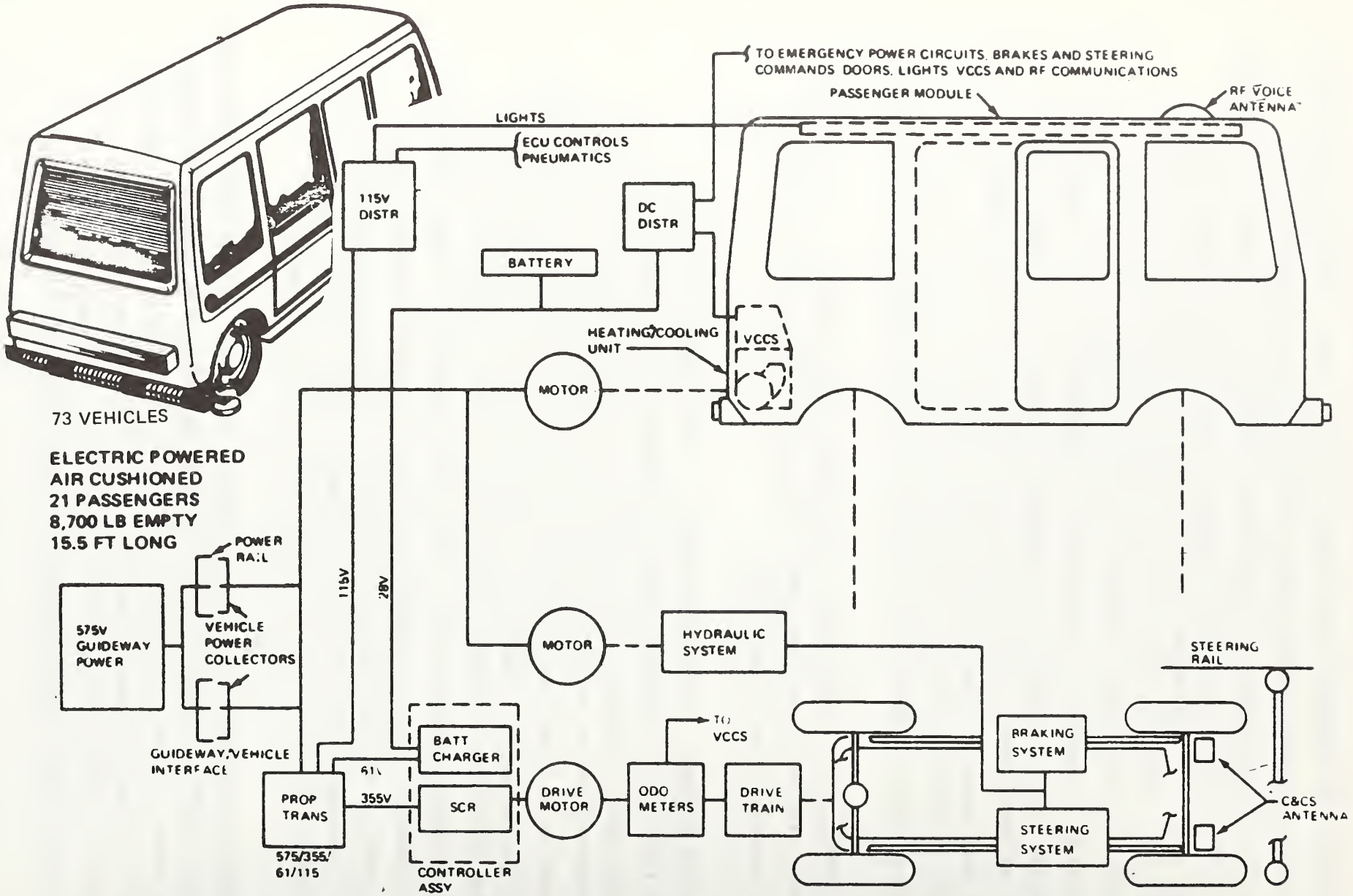


FIGURE 2-1. MPM VEHICLE FUNCTIONAL SCHEMATIC

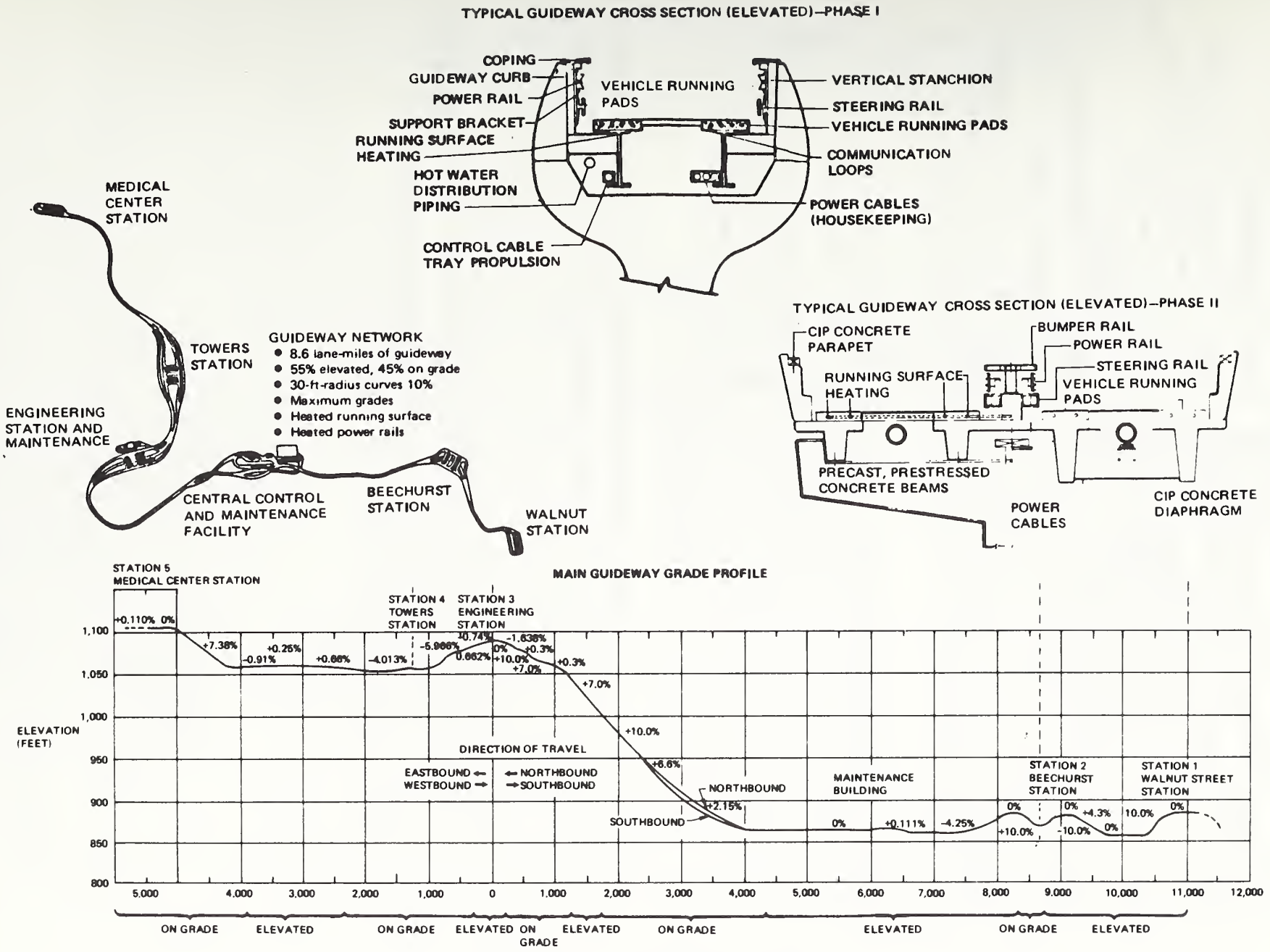


FIGURE 2-2. GUIDEWAY SYSTEM

main guideway structure is a double guideway with two-way travel. The guideway and power rails are heated for ice and snow removal to enable safe operation of the vehicles during winter. The station facilities provide for handling the passengers, directing them to the proper platform, and loading and unloading the vehicles. The stations are two level for passenger and vehicle movement on a non-interference basis. The power distribution system converts local input power at the main substation and distributes the power to all facilities and guideway power rails. The maintenance facility has a storage yard, a test track, and a maintenance building for servicing vehicles and C&CS equipment. The central control and communications facility is co-located with the maintenance facility.

Control and Communications System. The primary purpose of the C&CS is to automatically control and monitor the entire MPM system under the supervision of the central operator. This includes control and monitoring of vehicle dispatches and vehicle operations on the guideway and in the stations. The C&CS provides communications between central control, stations, vehicles, and maintenance facilities.

The C&CS is divided functionally into three major control and communications subsystems as shown in Figure 2-3: Central (CCCS), Station/Guideway S/(GCCS), and Vehicle Control and Communications Subsystem (VCCS). A functional diagram of the interfaces is shown in Figure 2-4. As can be seen in this diagram, the redundant computing system is part of the Central Control and Communications Subsystem and the Station Control and Communications Subsystem. The following sections describe the hardware and software which comprise the MPM redundant computing system.

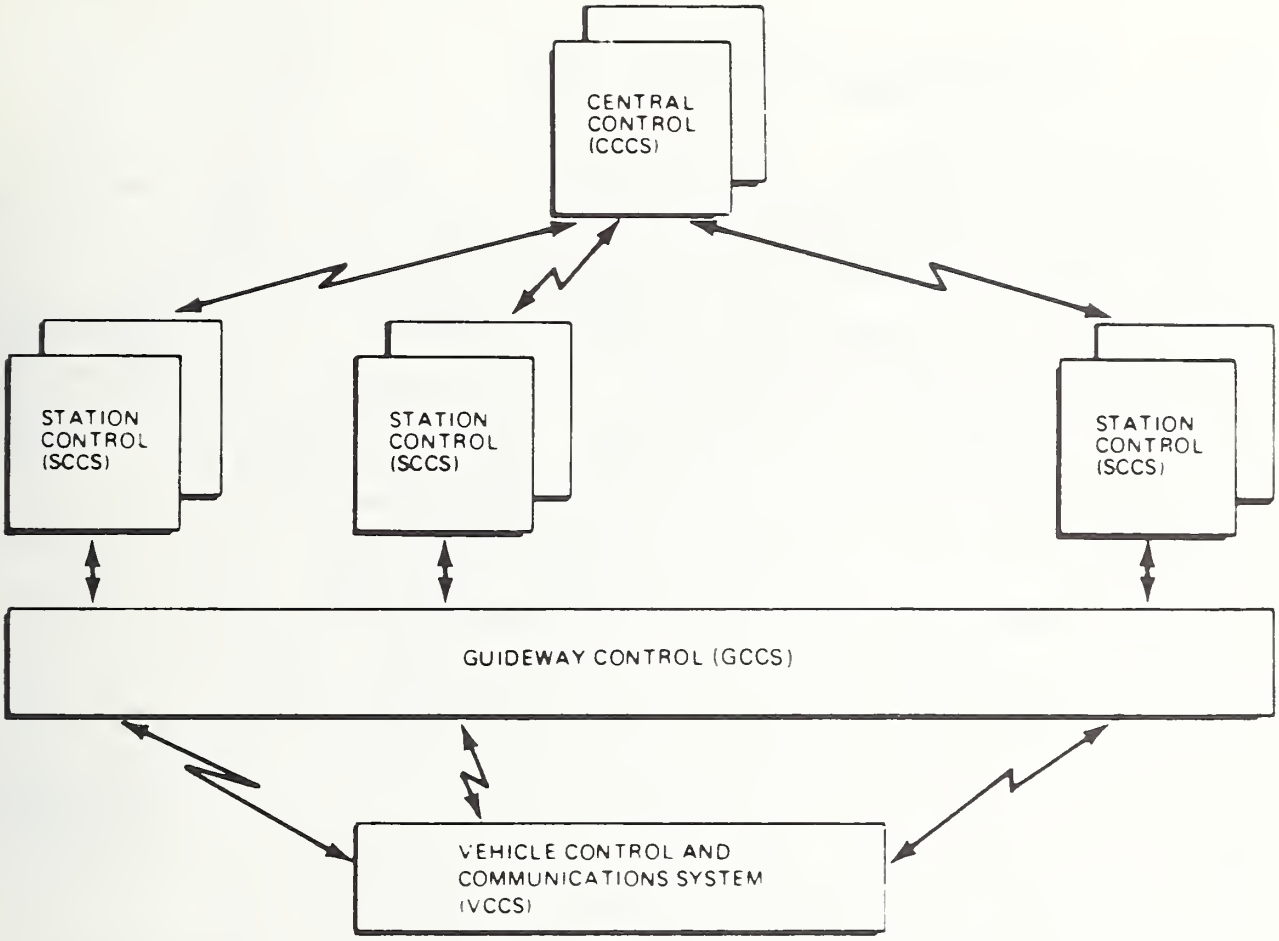


FIGURE 2-3. C&CS CONFIGURATION

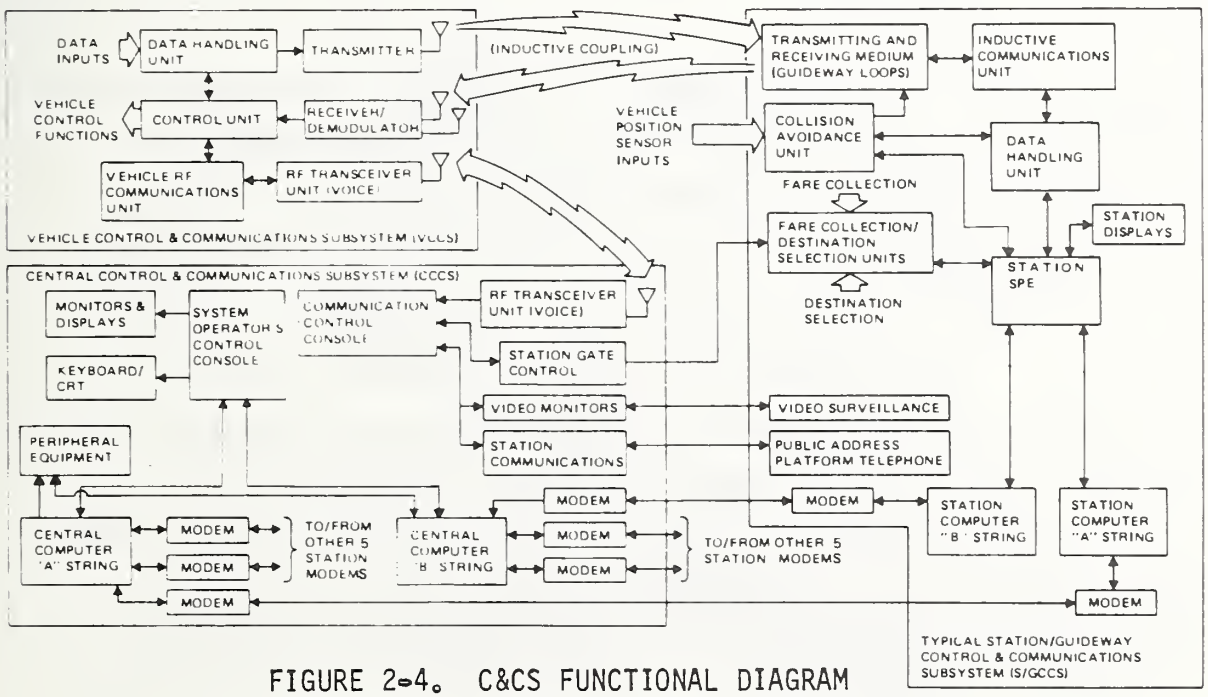


FIGURE 2-4. C&CS FUNCTIONAL DIAGRAM

2.2 COMPUTER HARDWARE SYSTEM DESCRIPTION

The MPM Phase II redundant computing system is a string redundant system consisting of a central control dual computer complex and six dual station computer systems. The six stations are geographically located along the 8.6 mile guideway and electrically connected to central through a redundant modem communication system. Figure 2-5 shows a functional diagram of the redundant computing system. The basic design concept was to develop a system using off-the-shelf commercial hardware to the maximum extent possible, thus decreasing the need for new design and reducing the requirement for special training of maintenance personnel. A string redundant hot spare with synchronized parallel processing was used as the redundancy approach. To add flexibility to selecting which computers made up a string, the communication links between central and each station computer were made switchable by a modem reconfiguration unit. The development of this design occurred over two phases of the MPM program. Most of this design resulted from implementation of the basic requirements, but some was the result of careful analyses of infrequent but undesirable failures noted during the tests and operations performed in the Phase I portion of the program. The requirements, the failures, and the details of the resulting design are presented throughout this report.

The central computer complex contains the man/machine interface equipment, data collection media, and processing equipment to coordinate overall system control. The central computer configuration is shown in Figure 2-6. The equipment consists of the following standard equipment found in most computer complexes: processor, memory, disk systems, magnetic tape units, real time clocks, line printers, and modems with interface controllers. Also included are five special devices to interface with the MPM unique man machine interface equipment and to control and monitor the redundancy scheme. The special interfaces consist of the following equipment: a cathode ray tube terminal with the keyboard key caps modified to fit the human requirements of controlling the system; a remote bootstrap system allowing downloading of software to the station computers without assistance from any remote site; an

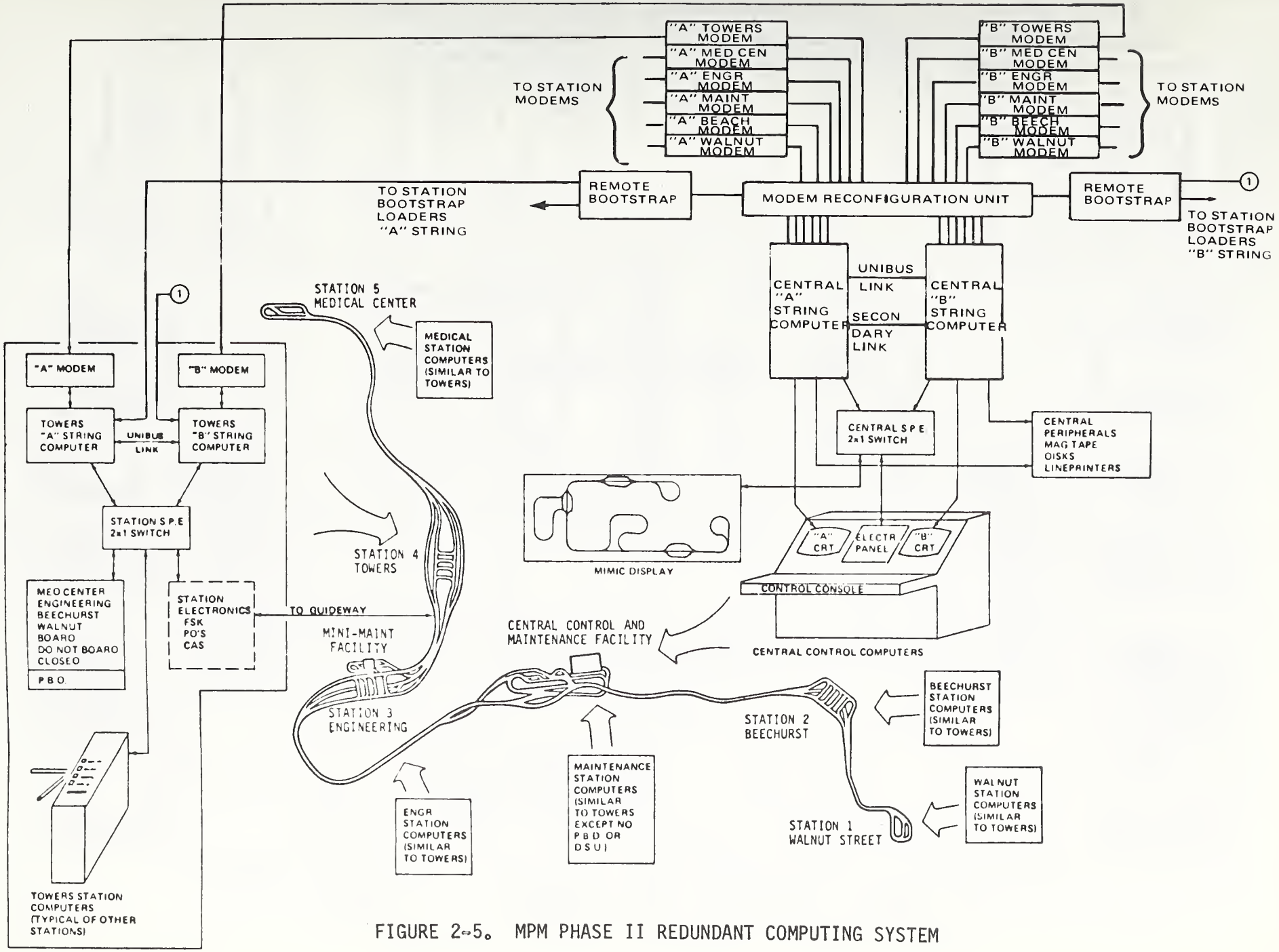


FIGURE 2-5. MPM PHASE II REDUNDANT COMPUTING SYSTEM

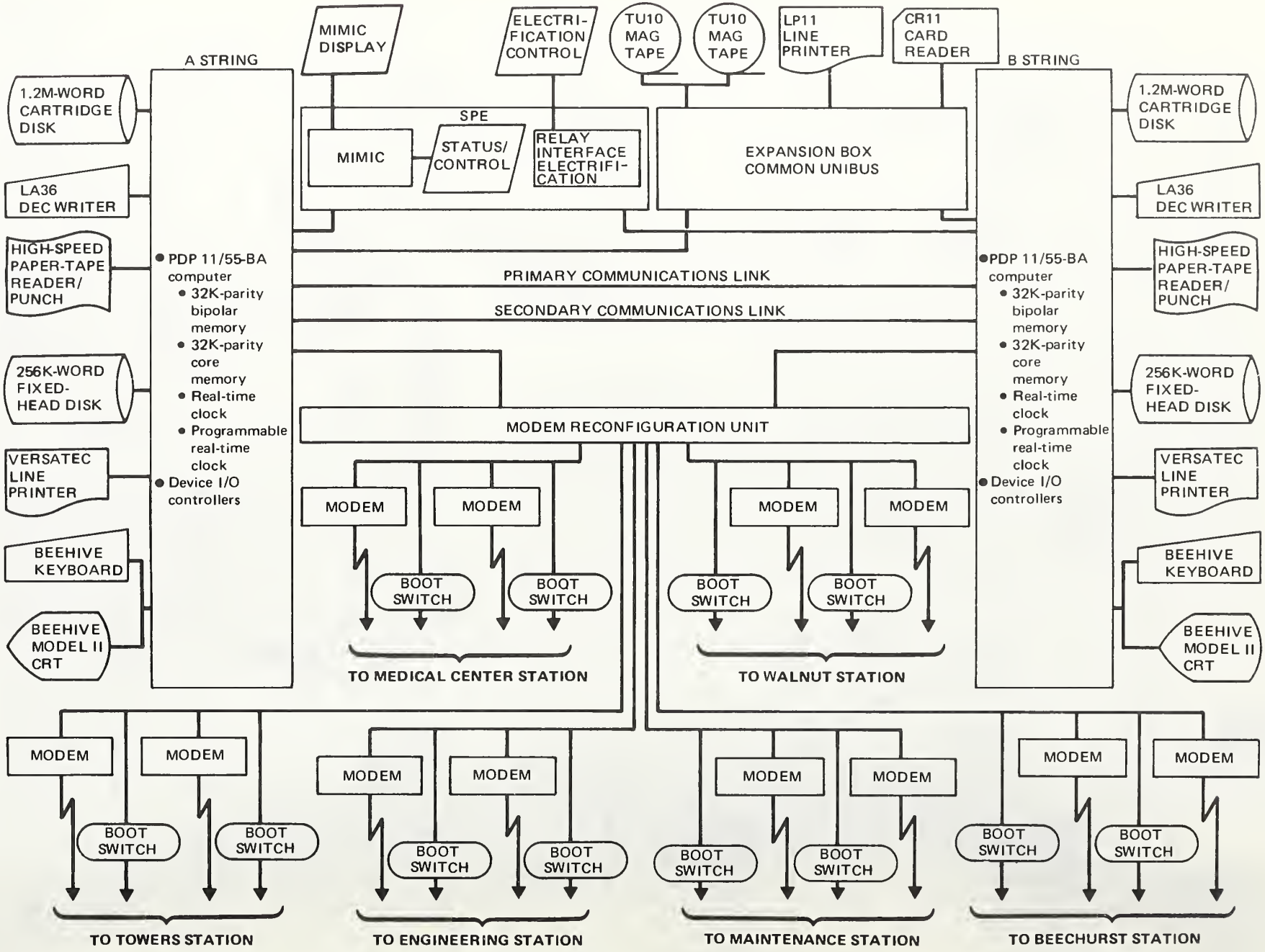


FIGURE 2-6. MPM CENTRAL COMPUTER CONFIGURATION

inter-computer communications link with a secondary backup provides the communication necessary to provide redundancy management; a modem reconfiguration unit allowing station computers to be manually switched between the A and B strings; and a central special purpose equipment (SPE) unit allowing automatic software switching so that only the prime computer outputs are gated to a mimic board. The mimic board is a display with a functional model of the guideway along which are lights representing segments of the guideway as divided for the collision avoidance system and the station berth areas. The hardware/software system illuminates the lights for segments which are occupied with a vehicle enabling the operator to monitor overall vehicle motion. Also connected through the central SPE is a wire "or" electrification system trip interface to provide vehicle propulsion power removal as required to ensure passenger safety.

The station computer systems contain equipment required to interface with the guideway control electronics and the passengers. A typical station computer configuration is shown in Figure 2-7. Each of the station computer systems consists of the following standard equipment: processors, memories, floppy disks, teletypes, real-time clocks, and modem communication equipment. The station computers also contain general purpose interfaces for interfacing to the guideway control electronics and the passenger-machine interface electronics. These general purpose interfaces are typical for all stations except that maintenance station does not include any passenger-machine interfaces. These interfaces in each of the dual computers are connected to the single thread guideway and passenger electronics through the station special purpose equipment (SPE).

There are three basic types of SPE control interfaces used to interface with the guideway and passenger equipment at the stations. One type is used for input of data from the destination selection units (DSU), the presence detectors (PD), and the vehicle to guideway FSK downlink messages. The second type is used for output of the collision avoidance safetone loop control and for the output of vehicle uplink messages. The third is used for control of the passenger boarding displays.

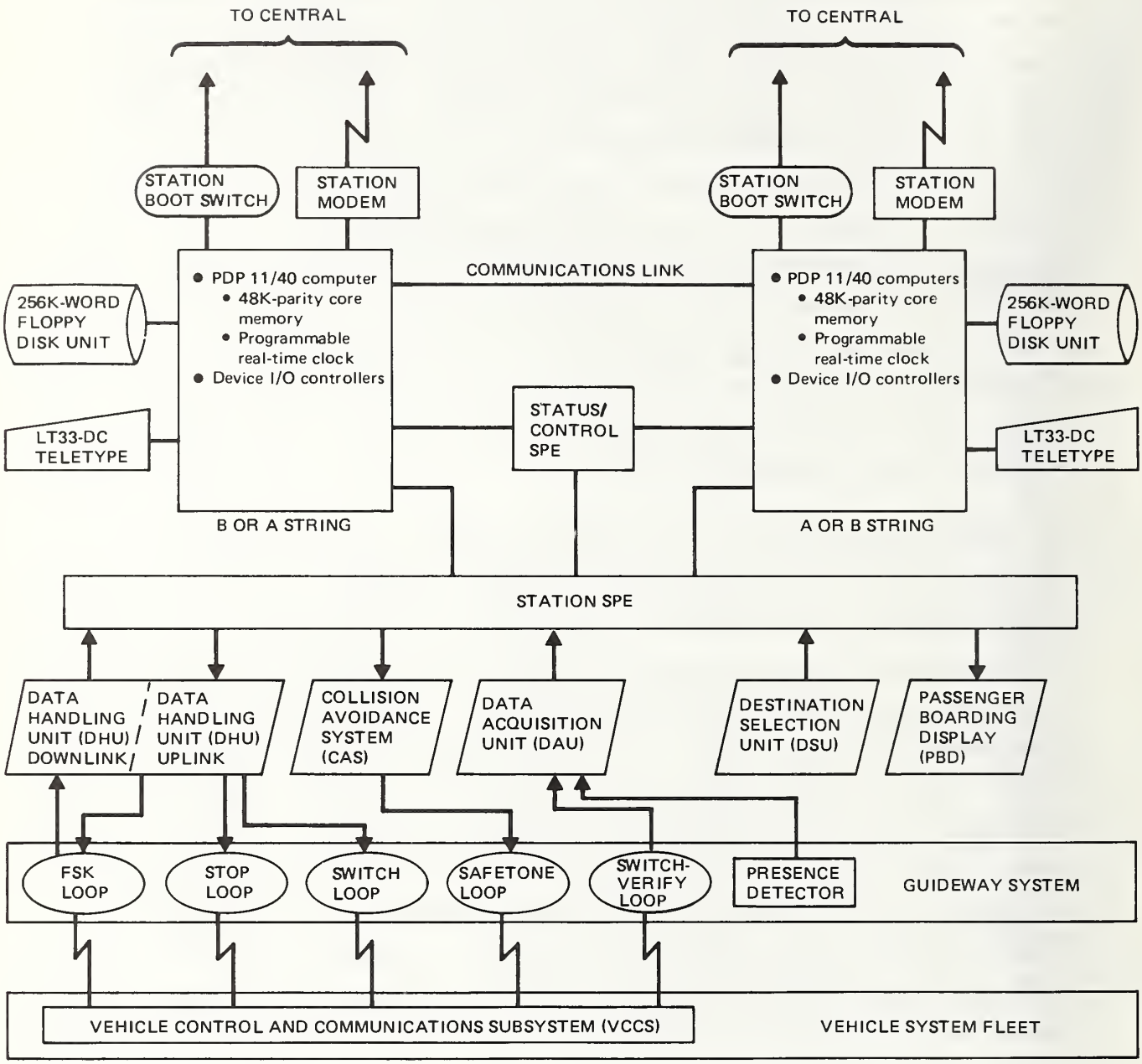


FIGURE 2-7. MPM TYPICAL STATION COMPUTER CONFIGURATION

The specially designed station SPE serves as the transition point from the off-the-shelf dual computer network to the single string station electronics. For the input devices the SPE provides handshake controls, and data routing, holding, gating, and timeout functions required for asynchronous data input to both the dual computers. For the DHU uplink output the SPE provides handshake controls and data routing so that only the prime string outputs are gated to the guideway system. To maintain data processing symmetry the DHU uplink interfaces in the backup computer string are interfaced with logic in the SPE which simulates the data transfers to the guideway system. The collision avoidance outputs do not require any handshake controls, but, again, the SPE only gates the prime string outputs to the guideway system. The passenger boarding display is simply a wire "or" since the control of these signs does not require exact synchronization.

It is interesting to look at the operations the SPE must perform to provide input to two computers which will answer the data inputs asynchronously. When in dual computer mode, the SPE places the data on the input lines for both computers and then generates an interrupt to both computers simultaneously. When the fastest computer acknowledges this interrupt, two functions are performed: first, a holding flip-flop is set indicating the interrupt has been answered but that the signal has not passed to the external device; and second, a timeout monitor is started. When the slower computer acknowledges the interrupt, a second holding flip/flop is set; the logical "and" of these two holding flip flops generates an acknowledge signal which passes to the external device and clears the timeout monitor and both holding flip-flops completing the input cycle. If the slower computer does not acknowledge the interrupt before the timeout monitor expires, the following occurs; generation of a timeout pulse which sets an error flip-flop, generates an acknowledge signal which passes to the external device and clears both holding flip-flops. The error flip-flop then inhibits further timeout monitor and the dual mode terminates. The error is cleared by the operating (faster) computer after the SPE is switched to single string mode. When in single string, the handshake and data from the operating computer pass directly to the external device, and all dual string functions and timeout monitoring are disabled. Detailed circuit descriptions

and design requirements of the redundant computing equipment is treated in later sections of this report.

2.3 Computer Software System Description

The operational software is a subsystem within the C&CS which controls the system configuration, manages the movement of vehicles and passengers between stations, and controls the movement of vehicles on the guideway and in the stations. The operational software consists of four major programs: Central Application Program (CAP), Passenger Station Application Program (PSAP), Maintenance Station Application Program (MSAP), and the Executive Program (EXEC). By combining the executive with each of the other three programs, three software segments are created as shown in Figure 2-8. Each segment resides in a computer at an assigned location: central segment in the central computer, passenger station segment in the five passenger station computers, and maintenance station segment in the maintenance computer. The passenger station segment software code is identical for each station with unique parameters contained in each data base to accommodate the differences in station configurations. (The phrase "Station Application Program" (SAP) refers to the application programs in both the passenger stations and maintenance station.) Communication between segments is accomplished through messages transmitted over the modem communication lines. The software in the redundant strings is identical. The same software image is loaded on the A string and on the B string. Thus, there is a central segment in each of the two central computers, a maintenance station segment in each of the two maintenance station computers, and so on. Communication between segments at the same location is through control signal and data transfers over the bus links connecting each redundant computer to its companion computer.

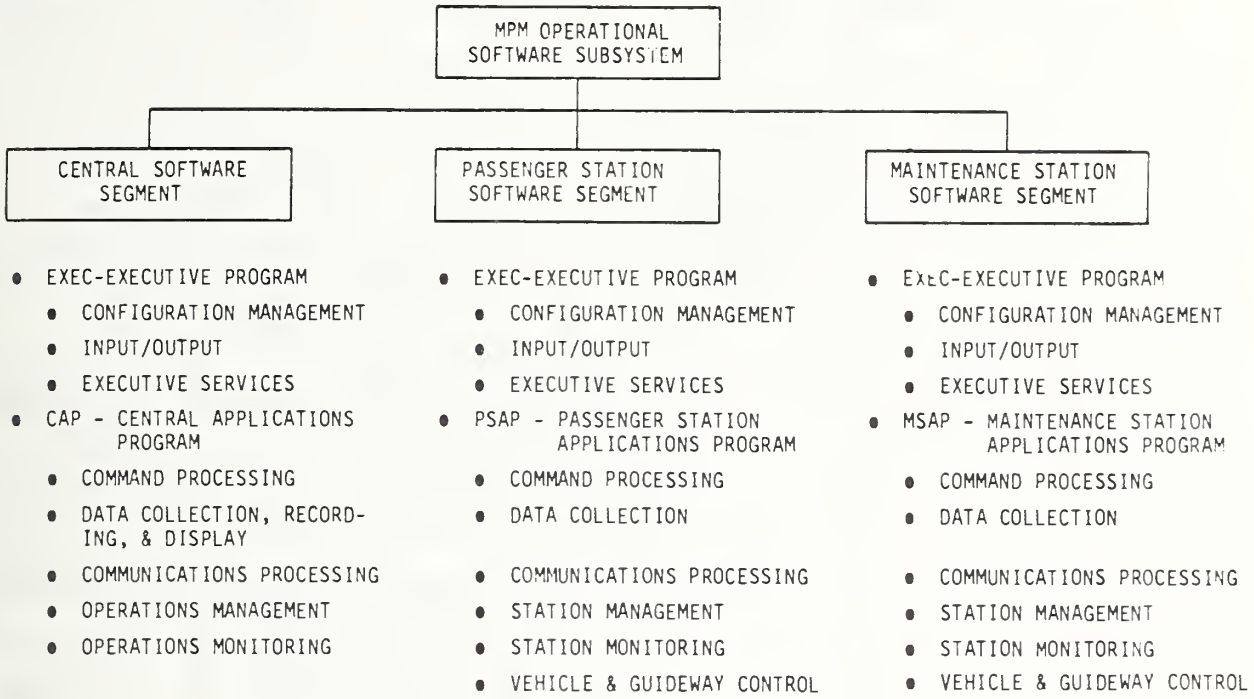


FIGURE 2-8. MPM OPERATIONAL SOFTWARE SUBSYSTEM ORGANIZATION

Each computer in the network contains an applications program and an executive program. The applications programs perform functions which control system operation from the passenger and operator point of view. The executive program controls the processing performed by the applications programs and provides the software interface with the computing system and external environment. The executive program is allocated the responsibility of managing the redundant computer system functions. The application software is nearly blind to the fact that a second string exists and whether the application software is in the prime or backup computer string. The redundant functions of the executive program are described in detail in this report. A top level description of the application software is included to provide an overview of the functions performed by the operational software in the operation of the system.

Operational Software Organization Levels. The organizational levels for the Operational Software Subsystem are indicated in Figure 2-9. As indicated previously, this subsystem consists of three segments,

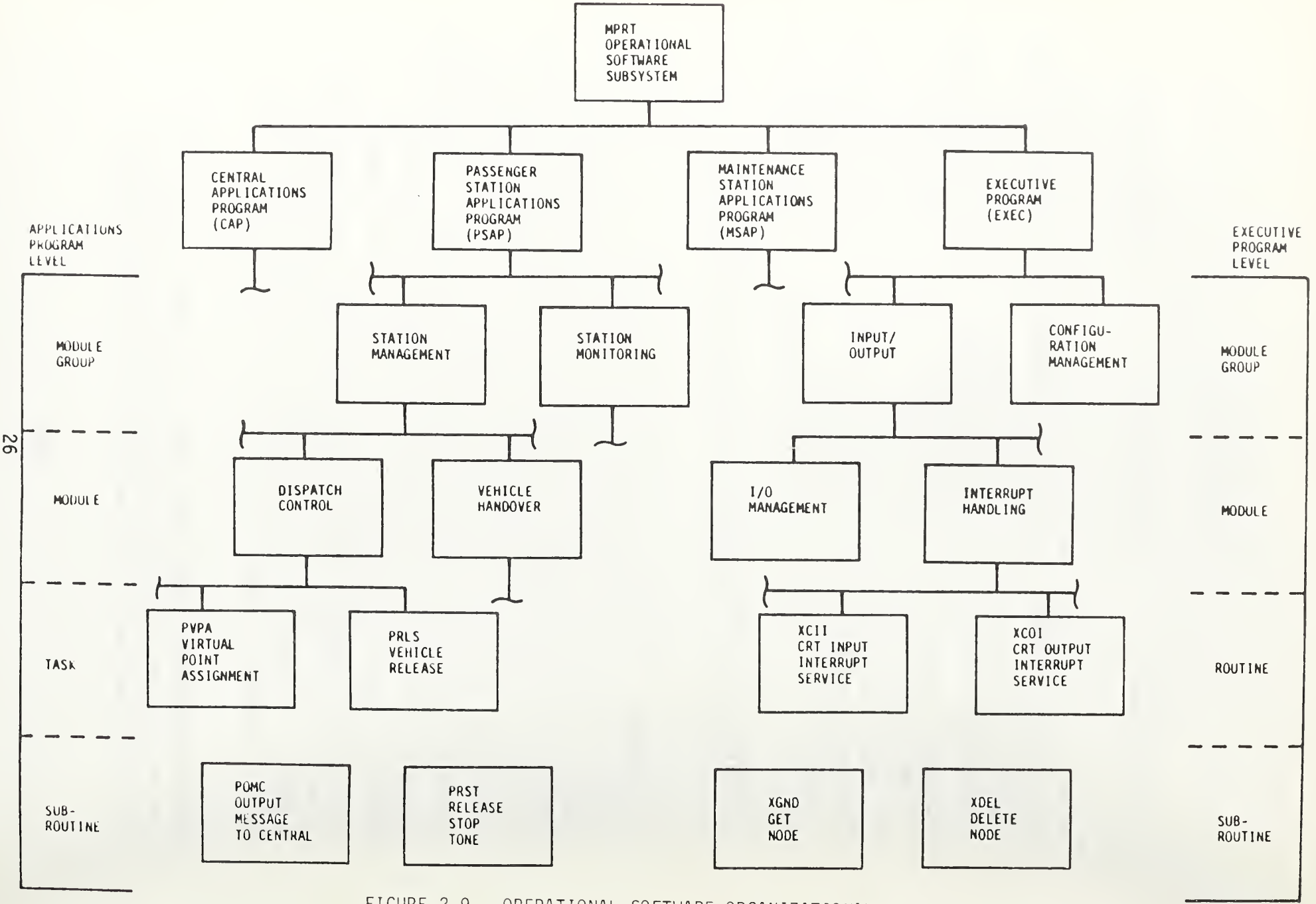


FIGURE 2-9. OPERATIONAL SOFTWARE ORGANIZATIONAL LEVELS

each segment having an executive and an applications program. Each program is organized hierarchically into sub-elements. The applications program levels are module group, module, task, and subroutine. Tasks are the entities whose interfaces are accommodated by the executive, for example, scheduled for execution. The executive program levels are module group, module, routine, and subroutine. Some executive routines contain tasks. Subroutines are commonly used processes which are called by tasks and routines.

Central Applications Program (CAP). The CAP software is the applications program which resides in the central computer. CAP provides the central operator interface and the centralized application control and monitoring functions. CAP records system operational data, displays and logs system status data, and executes operator commands. CAP maintains the vehicle motion time base (virtual points) and coordinates and communicates with the station software to effect the distribution and movement of the vehicle fleet throughout the system.

The CAP software consists of five module groups as shown in Figure 2-10. The Command Processing module group enables system startup, accepts operator inputs from the central operator CRT keyboard, and executes operator commands.

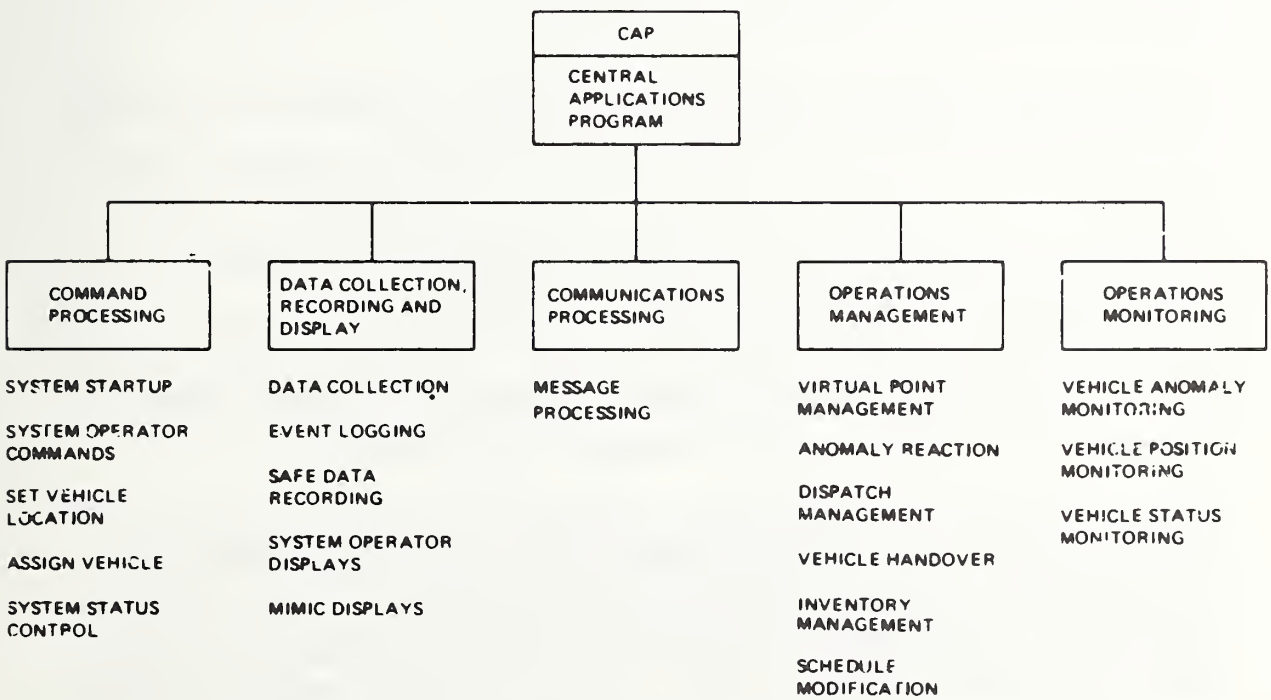


FIGURE 2-10. CENTRAL APPLICATIONS PROGRAM ORGANIZATION

The Data Collection Recording and Display module group provides the following:

- Data collection - includes output of runtime information such as vehicle mileage to magnetic tape.
- Event logging - provides for printouts of vehicle locations, miles, and enabled hours and real-time system fault logging.
- Safe data recording - provides periodic output of vehicle status information to cartridge and fixed-head disks.
- System operator displays - outputs system status and current anomalies to the CRT, along with the last requested display. Anomalies and displays are printed upon operator request.
- MIMIC display - periodically updates the mimic board to reflect current vehicle locations.

The Communications Processing module group accepts messages from the Passenger and Maintenance stations applications programs and queues the proper CAP task to process the messages.

The Operations Management module group provides for dispatch of vehicles and for maintenance of the virtual points and subpoints which it uses to synchronize dispatches. It also provides for management of vehicle inventories in demand mode, initiation of dispatch rate changes in schedule mode, and reaction to anomalous conditions.

The Operations Monitoring module group maintains the current status of each vehicle in the system as reported by the stations.

Passenger Station Applications Program (PSAP). The PSAP software is the applications program which resides in the various station computers. PSAP controls the movement of individual vehicles on the main guideway within the control zone of a single station, on station merge and demerge ramps, and in station channels. This includes ramp and channel switching

and door control at load and unload berths. PSAP also accepts passenger destination requests, controls passenger boarding displays, and communicates with central to coordinate timing and vehicle movement. PSAP also receives commands and reports station and vehicle status.

The PSAP software consists of six module groups as shown in Figure 2-11. The Command Processing module group supports processing for commands and data originating from system operator keyboard and from CAP.

Data Collection consists of one module which supports the collection of vehicle trip and passenger request data for recording and display by CAP.

The Communications Processing module group accepts messages from CAP, the data handling unit (DHU), and the destination selection unit (DSU) and initiates the appropriate processing of these messages. It also provides for the preparation of vehicle messages to be transmitted through the DHU and for the illumination of appropriate passenger boarding displays.

The Station Management module group provides for the updating of virtual points, reaction to vehicle and station electronics anomalies, dispatching and handover of vehicles, and the assignment of vehicles to provide passenger service.

The Station Monitoring module group monitors vehicle reported faults, detects other anomalous conditions of the vehicles, monitors all vehicle positions within the station control zone, and verifies proper vehicle switching for vehicles entering or by-passing a station.

The Vehicle and Guideway Control module group controls vehicle switching on the guideway and within station ramps and channels, controls vehicle movement and door operations within the station channels, verifies proper vehicle merging onto the main guideway, and controls the software collision avoidance system (CAS).

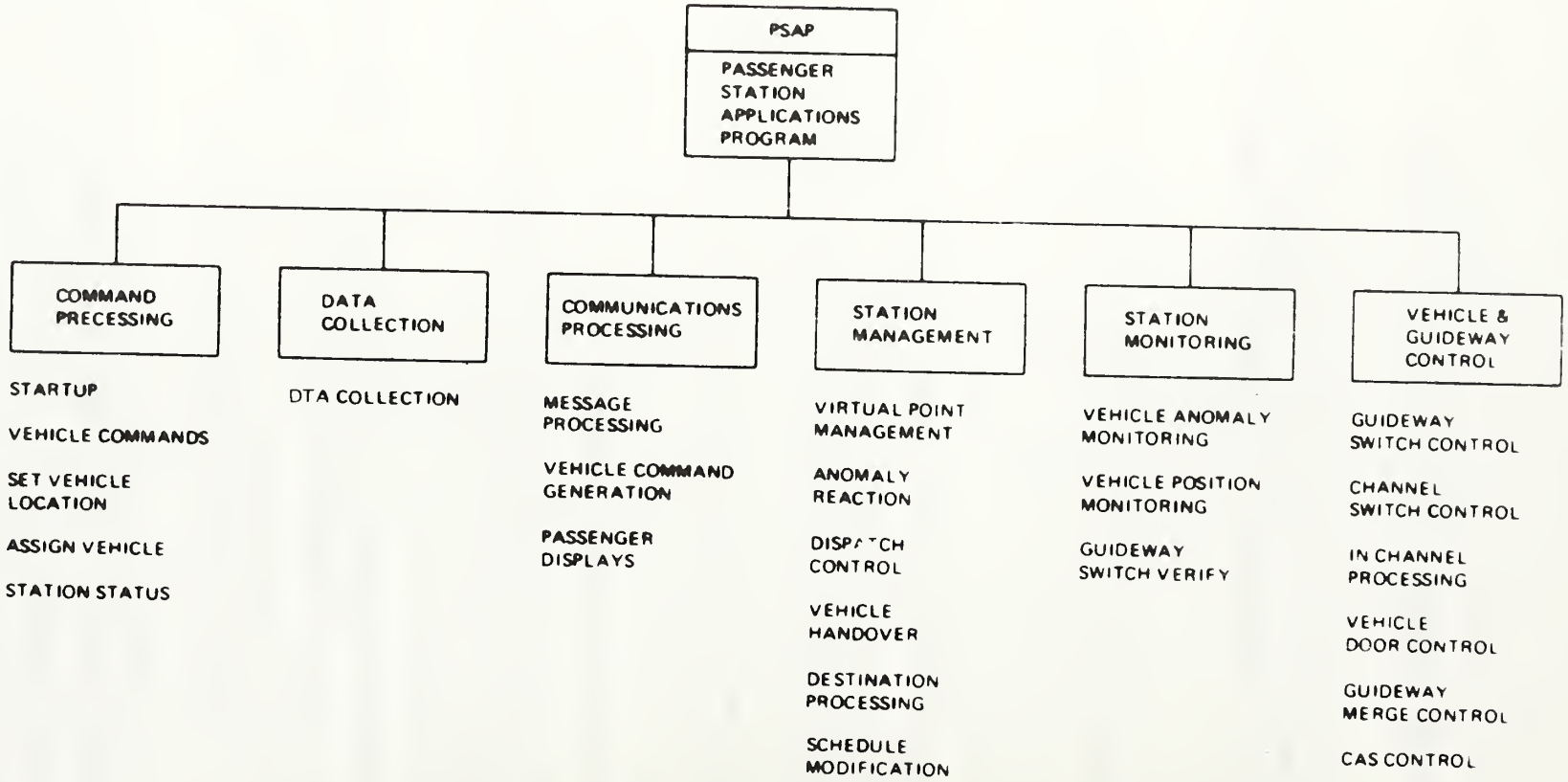


FIGURE 2.11. PASSENGER STATION APPLICATIONS PROGRAM ORGANIZATION

Maintenance Station Applications Program (MSAP). MSAP functions are similar to those of PSAP in providing vehicle control and communications with central. In addition, vehicle control is provided for a test loop at maintenance which is used for checking vehicle operations and performance. There is no passenger interface at maintenance.

The MSAP software consists of six module groups as shown in Figure 2-12. The composition of the groups is similar to that described for PSAP. The differences are noted below.

The Command Processing module group supports processing of commands to control the looping of vehicles on the maintenance test loop in addition to the other functions described.

The Communications Processing module group performs the same functions as this module group in PSAP except that it is not required to process DSU inputs or to output data to the passenger boarding displays since those devices are not included in the Maintenance station environment.

The Station Management module group performs the same functions as this module group in PSAP with the exception of the assignment of vehicles to provide passenger service.

In addition to the functions listed for the PSAP Station Monitoring module group, MSAP also monitors vehicle switching on the maintenance test loop.

The Vehicle and Guideway Control module group includes control of test loop switching and monitoring of merge positions on the test loop in addition to the functions described for PSAP.

Executive Program. The executive program is the real-time control operating system software for the MPM operational software. Three versions of the executive exist: one for the central PDP-11/55, one for the passenger station PDP-11/40, and one for the maintenance station PDP-11/40. All of the versions , regardless of which computer they reside in, are functionally the same to the first level of detail.

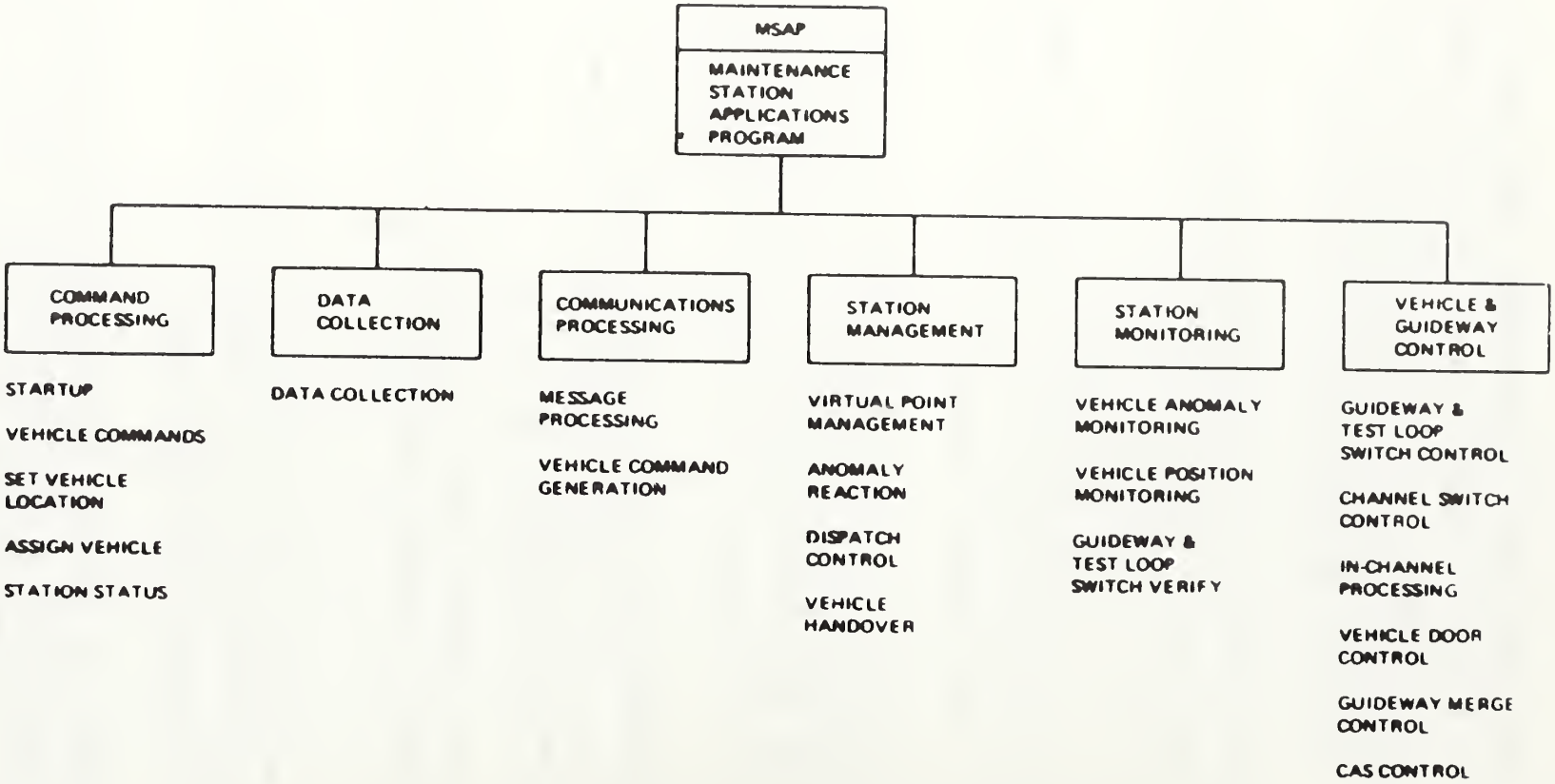


FIGURE 2.12. MAINTENANCE STATION APPLICATIONS PROGRAM ORGANIZATION

However, the System Status Monitoring, Reconfiguration, and Configuration Control functions differ for the station and central computers in detail implementation. The station and maintenance versions differ only in that the station executive supports the passenger interface devices which do not exist at maintenance.

"Executive" is used to identify any of the executive versions which are incorporated in the MPM computers. Whenever a distinction is to be made, the terms "central executive", "maintenance executive", and "station executive" will be used. Unless explicitly stated otherwise, "station executive" refers to both the station and maintenance executive.

The executive program provides the operation and computing environment for integrating the various hardware devices of the redundant, distributed computing system and the application programs into a unified and viable control and communications system. The executive is blind to peripheral device and application program function performing the role of message passer and task dispatch manager. The executive raises the level of the computers and computer network such that application programs can be developed and executed at a higher and more readily understandable level. For example, the executive handles the detailed machine level I/O functions allowing the applications simply to request input and output.

The executive program consists of three module groups: Executive Services, Input/Output, and Configuration Management. The executive program organization is shown in Figure 2-13. The executive module groups and modules are numbered in Figure 2-13 to provide identification in later discussions.

Executive Services module group consists of the Dispatcher, Scheduler, Executive Service Requests, and FORTRAN Object Time System (OTS) Interface modules. The Dispatcher selects the next task to start execution (be dispatched) by scanning a linked list (dispatch queue) of tasks which are ready to execute or are waiting for the completion of an event.

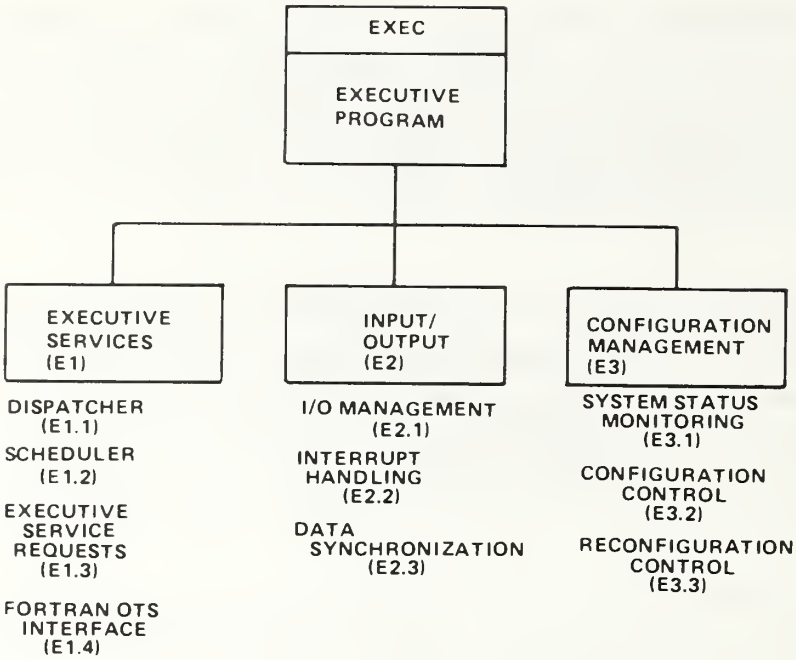


FIGURE 2-13. EXECUTIVE PROGRAM ORGANIZATION

The highest priority task which is ready to execute is selected and control is passed to that task. The Scheduler maintains a linked list of tasks (the schedule queue) which are scheduled to be executed at some time in the future. The Scheduler supports both one time only executions and periodic (repeating) executions. The Executive Service Requests module consists of a set of commonly used functions which are requested by the applications software and performed by the executive. The repertoire of services available enable tasks to interface dynamically with other tasks, to interface with external devices, and to request information from and to pass information to the executive. The FORTRAN OTS Interface provides the interface between the FORTRAN IV run-time routines and the applications and executive tasks. The OTS interface defines which run-time routines are to be extracted from the OTS library during system generation and, therefore, defines the FORTRAN features which will be supported by the executive. The FORTRAN OTS Interface also reports detected FORTRAN run-time errors to the executive system status monitoring function.

The Input/Output module group consists of the I/O Management, Interrupt Handling, and Data Synchronization modules. The I/O Management module performs the initiation and termination of input and output processing for the various computer interface devices. The Interrupt Handling module processes device interrupts, performs the processing necessary to continue device data transfer, and notifies I/O Management when the data transfer has been completed. The Data Synchronization module performs the transfer of variable data between the prime and backup computing strings enabling the synchronization of processing activities in the separate strings.

The Configuration Management module group consists of the System Status Monitoring, Configuration Control, and Reconfiguration Control modules. The System Status Monitoring module provides for the detection of failures in the computer hardware and software systems. When System Status Monitoring detects a failure, it notifies the Reconfiguration Control module. The Configuration Control module consists of two functions: computer memory configuration and computer system load and initialization. The computer memory configuration function takes the output of the system generation process and builds the operational software system disk image. Configuration directives to this function are used to generate computer memory images for each of the computers. These images are stored on the system disk along with directives for use by the system load and initialization function. The load and initialization function uses the directives on the system disk to load the operational software into the proper computers. This function also initializes the computer hardware and initiates processing for the central and station executives. The Reconfiguration Control module reacts to failures detected by the software (and computing hardware). Reactions to failures include reporting of failures to be logged and reconfiguration switching for those failures requiring a configuration change.

Operational Software Concepts. The operational software is a real-time control system. The software is event driven rather than cyclic and asynchronous rather than synchronous. This provides a more responsive control system and one which is better able to handle peak load conditions

without the frame overrun problems of cyclic software. The software supports the dual string distributed processing network with a controlling prime string and a hot backup.

The same software is loaded in each string, and either string can be the prime string. The software makes decisions based on prime or backup status. There are no tests in the software to determine whether it is executing on the A or B string. The software in the two strings run independently, each string making its own decisions based on its own inputs and calculations. The backup string is synchronized to the state of the prime string in a "point" synchronization. That is, the backup is data synchronized to the prime string at a point in time and then stays in relatively close synchronization since both strings are presented the same inputs. The two strings are not in precise synchronization in the sense that they do not execute the exact same instructions at the same time. However, as long as the two strings output the same commands within a short time of each other they are said to be in synchronization.

A description of the design of these redundant computing software functions is contained in Section 3.

2.4 Hardware/Software Functional Allocation

At the time the MPM computer system was designed to a dual string configuration, the functions to provide redundancy were allocated to the executive software and the computer hardware. The ground rules which guided this allocation were: reduce the need for special built hardware by using as much off-the-shelf hardware as possible; reduce the redundancy allocations to as few subsystems and components as possible making the majority of the system believe that only a single computer string exists; minimize the hardware and software required to implement the redundancy; and make the hardware/software allocation as "natural" as possible by following prudent design practices. The following identifies which functions were allocated to hardware and which to software.

2.4.1 Functions Performed by Hardware in Redundant Elements

The basic hardware design approach in implementing the MPM redundant computing system was to develop a system in which all interfaces to the external system devices appeared to the external devices as if they were a normal single computer interface even though the redundant system can assume the following modes:

1. "A" string computers operating alone.

2. "B" string computers operating alone.

3. "A" string in control with all I/O synchronized so that "B" string is mimicking "A" string in real-time and can assume control without I/O interruption.

4. "B" string in control with all I/O synchronized so that "A" string is mimicking "B" string in real-time and can assume control without I/O interruption.

To accomplish this the SPE hardware provides the following redundancy functions:

- Logic to select which computer string's data is presented to the external system devices;

- Logic to provide synchronization of the interface timing and control signals when operating in dual mode; (this consists of "anding" and holding logic to provide for the temporary holding of one computer acknowledge signal until the companion computer provides its acknowledge before the acknowledge passes to the external device and watch dog timeout logic to assure that the delay between the redundant elements does not exceed the latency requirements of the external system;)

- Mode selection logic so that the software can select /control which of the four modes is in effect;

- Interrupt logic to allow the station SPE equipment to notify software of redundancy timeout errors.

The following functions are performed by hardware other than the SPE hardware to support the redundant computing scheme:

- provision of communication equipment allowing the software a means of system monitoring for redundancy control, synchronization, timeout and error monitoring, and reconfiguration control;

- self-check monitoring for the internal computer operations, such as memory parity and memory management;

- remote startup and restart without affecting in any way system operations.

2.4.2 Functions Performed by Software in Redundant Elements

The executive software commands the central and station SPEs to the proper mode for a given system configuration. Upon system startup the prime string, the first to be loaded, commands the central and station SPE to prime single mode. When the backup is loaded, the prime commands the station SPEs to dual mode. When either string seizes control of the system upon a failure in the other string, the software in the seizing string sets the central and station SPEs to prime single mode.

The software synchronizes the backup string to the prime's operational state by transferring all variable data in the prime string to the backup string via the intercomputer bus links. Before the strings are synchronized, the backup string software ignores the content of system inputs, such as vehicle downlinks, since it has no knowledge of system state. Once the software synchronizes the backup string to the prime, both strings must receive the exact same inputs in order to maintain synchronization. At the stations the inputs are presented to both strings by the station SPE hardware. At central each string has its own system operator CRT. The executive passes the prime CRT input characters to the backup string via the central bus link. The

executive discards the CRT input characters on the backup. However, each string outputs to its own CRT. The character routing is provided by the executive software and requires no special hardware. This function is totally transparent to the application routines which process the CRT characters.

The executive performs system status monitoring by monitoring the computer network for hardware failures. The executive maintains constant central to station and station to central communications. The executive performs other-central monitoring by maintaining constant central to central communications. The executive times out the central/station and central/central communications (i.e., failure to respond in a certain time period constitutes a failure). Reconfiguration decisions are made by the software when the software detects failures or when the hardware notifies the software of a component failure. The software performs reconfigurations on critical failures by performing a switchover to the backup string, or the prime string may seize control in prime single mode.

3. REQUIREMENTS AND DESIGN

This section identifies the functional requirements for the design of the redundant computing system. It describes the design of the redundant computing system hardware and software elements and emphasizes how the design satisfies the identified requirements. In addition, it discusses the off-the-shelf availability of components and the requirements for special design features.

The purpose of this section is not only to describe the requirements and design of the MPM redundant computing system, but also to show the types of features and capabilities which must be considered to implement redundancy functions of a dual redundant, hot-backup, real-time computing system. These considerations should provide guidance to anyone considering implementing or purchasing a redundant computing system.

3.1 Requirements for the Design of Redundant Computing System Elements

The MPM system has four major categories of requirements: conveyance dependability, safety, operability, and maintainability. Every requirement can be derived from one or more of these categories. There is no end to the range of possible system designs to make a system more dependable, safe, operable, and maintainable. However, these system designs must be evaluated on the bases of least cost and least risk of implementation to achieve the required level of performance. The purpose of this report is to describe and assess the MPM redundant computing system not to determine whether the MPM system is the optimum design approach. Obviously, there is a non-zero probability that a failure will occur in the remaining prime string before a previous failure can be repaired in the backup string. This situation can cause a system shutdown because the computer system is allocated safety critical functions. The system reenters operation as soon as a good string is assembled either by switching failed computers from string to string or, in the worst case, by repairing one of the two failures. It may be that a hierarchical

design in which component failures cause degradation in system performance rather than system shutdown would provide a more dependable system. However, a hierarchical system design would require more command and control equipment and cost more to implement than the MPM design approach. Regardless of the approach taken, safety critical functions must be allocated to some subsystem and they must be implemented in a fail-safe manner. Systems cannot be allowed to operate with a safety critical function inoperable.

It was determined that the MPM system with the redundant computing system described in this report would meet the required system availability for the least cost. It is on this basis that the requirements for the computing system are presented below.

3.1.1 General Requirements

The total MPM system is required to provide a level of safety such that there is no more than one accidental passenger fatality per 28 years (approximately). The redundant computing system is allocated a vital role in maintaining passenger safety. In fact, the most important requirement of the redundant computing system is the safety critical task of monitoring the system for passengers forcing open vehicle doors and getting on the guideway where they could come into accidental contact with the guideway power rail or be struck by a moving vehicle. The software is required to remove guideway power in the affected area upon receipt of an exits not closed report from the vehicle system. It is also required to remove guideway power when the capability to monitor the system for exits not closed is lost due to one or more failures. Thus, the computer system is an integral part of the system operation, and when the computer system cannot operate or monitor the system, it is required to shut it down in a fail-safe manner.

The Phase II overall MPM system is required to meet a target availability of 0.9700. Availability is defined as probability that the system is ready for use at any random point in time. This is equivalent to the ratio of system up-time to total scheduled operating time. Thus,

any system downtime reduces the availability. As a result of availability data collected during previous phases, each subsystem was allocated an availability requirement for Phase II. The redundant computing system was allocated an availability requirement of 0.9969 with an estimated MTBF of 175 hours and a mean down-time of 0.54 hours. This translates into less than $3.1X10^{-3}$ hours of downtime per operating hour or approximately 12.4 hours of down-time per year. This provided the reliability requirement against which to trade the design alternatives to determine the lowest cost system which meets this and the other requirements.

A very important requirement for the design of the redundant computing system is that to the greatest extent possible no single point failure can cause system downtime. Designing to this requirement also requires cost verses availability trades. For example, in the SPE there are single string components which would cause down-time upon failure. However, analysis shows that these components will fail so seldom that the target system availability can be achieved without the additional hardware and software cost to make these components redundant. Two or more simultaneous failures in the redundant computing system are of low enough probability that it is not required to consider them for availability purposes. However, automatic reactions (with operator backup) are required to maintain safety regardless of the number or coincidental occurrence of failures.

The redundant computing system is required to detect failures within itself and to reconfigure in a manner such that the failures do not cause system downtime. Switchovers to the backup system and seize controls by the prime system must not interrupt system operation or passenger safety. The only data which may be lost during a switchover to the backup is system statistical data which is output to the magnetic tape unit. The magnetic tape is shared and is switched between the strings via a manual bus switch. Switchovers and seize controls must be completed within 500 milliseconds after a failure which could interfere with system operation occurs. This 500 ms time requirement is derived from the collision avoidance system (CAS) at each station. The CAS

system is a checkin, checkout, block concept with checked redundancy between a hardware logic calculation and an independent software calculation. If a discrepancy between the software safetone "safe to proceed" output and the station electronics assessment of what the output should be exists for more than 500 ms, a CAS disparity occurs and the affected area is shut down to vehicle movement. Thus, the software must always be able to output its commands within 500 ms after a change in system state.

The backup system must be able to be brought into the system with no interference with system operation. This includes the loading, synchronizing, and arming of the backup system.

3.1.2 Hardware Requirements

This section identifies the requirements levied on the hardware design of that equipment needed to interface the two computer systems with single string external user devices to provide a redundant computing system. The MPM system utilizes eleven different types of user interfaces in addition to the controllers required to operate the redundancy interface equipment itself. To avoid repetition only one device interface together with the redundancy special purpose equipment (SPE) control requirements will be treated.

General Hardware Requirements

The general hardware requirements for the redundant computing system are as follows:

- Temperature shall be $10^{o}C$ to $50^{o}C$ while operating;
- Humidity shall be 20 percent to 95 percent;
- Design and fabrication shall utilize standard catalog modules as much as possible;

- The interface between the SPE and the user device shall be the same as if the equipment were interfaced directly with a single computer for all modes of operation of the dual interface equipment;

- The mode of operation shall be capable of being changed from dual to single at any time without effecting an interface data transaction;

- A symetrical command structure shall be provided so that the same software masks can be used in either computer;

- A diagnostic program to isolate logic failures or verify operation of all SPE equipment shall be provided;

- A maintenance panel shall be provided so that all interface controllers and SPE equipment diagnostics can be executed without opening drawers to install test cables;

- The logic associated with each single string shall be electrically and mechanically isolated with respect to power supplies, modules, etc., so that repair of one string can be accomplished to the maximum extent possible with the companion system operating the system;

- The SPE equipment shall include error monitoring and reporting logic which shall include a watchdog timer on all input transactions to verify that both computers respond to an input request within 50 ms of each other.

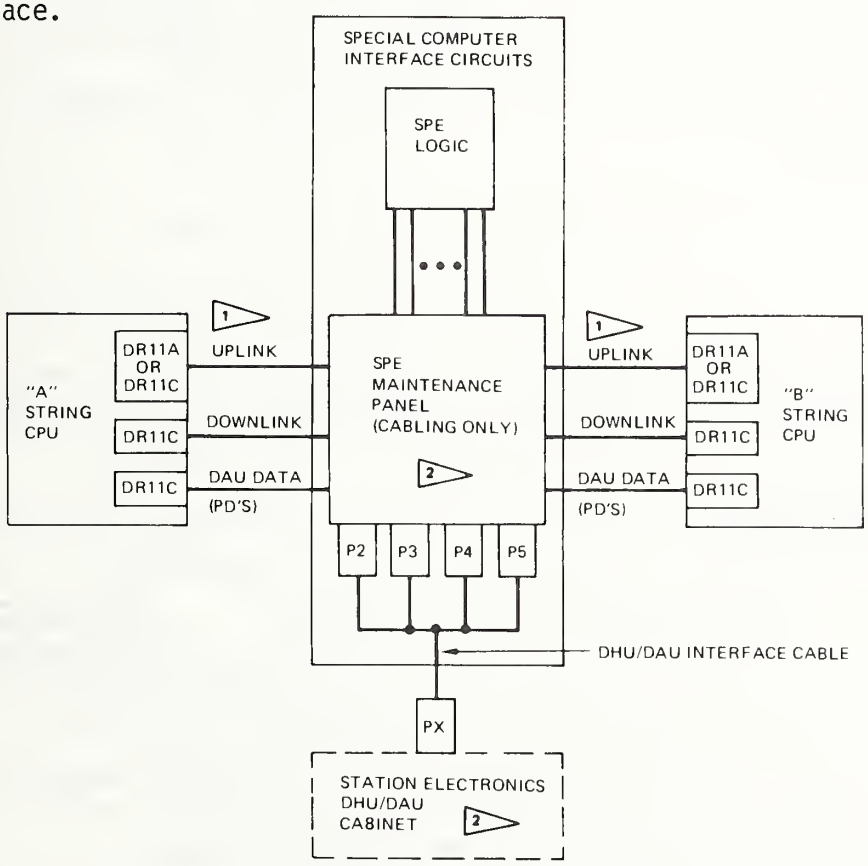
Specific Design Requirements. The redundant computing system is required to interface in a redundant role with the following devices:

Central Devices - mimic board, electrification trip, central secondary communications link, modem reconfiguration unit, remote restart, SPE control/status;

Station Devices - DHU uplink, DHU downlink, data acquisition unit, destination selection unit, passenger boarding displays, collision avoidance system, SPE control/status.

Each of these interfaces has specific hardware interface requirements which include cable pin assignments, logic levels, and timing relationships. To illustrate the content of these requirements, a portion of the dual computers to station SPE to data acquisition unit (DAU) interface requirements are presented here.

Computer-SPE-DAU Interface Requirements Functional Description. This interface inputs multiplexed information from the station electronics data acquisition unit (DAU). Figure 3-1 shows a block diagram of the interface.

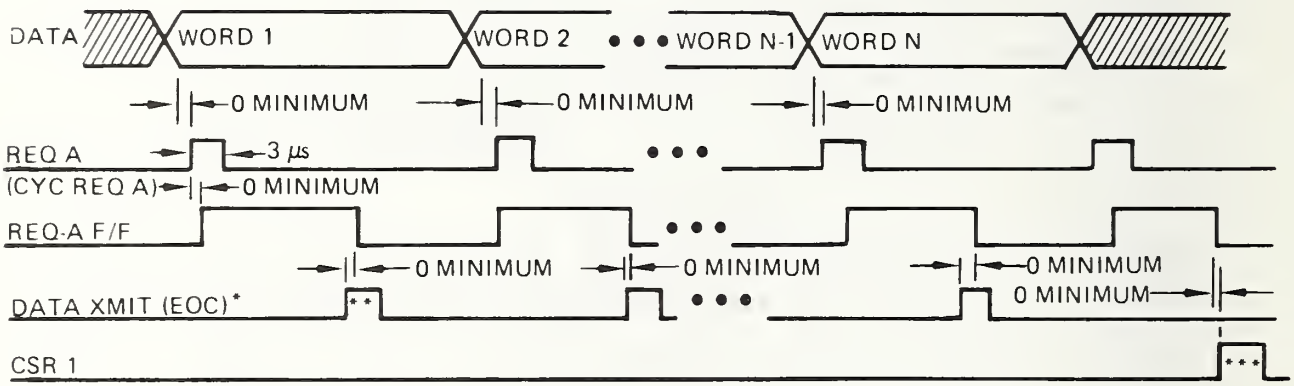


1 DR11A is used in Phase IB stations Walnut, Beechurst, Maintenance, and Engineering. DR11C is used in Phase II stations Towers and Medical Center.

2 Transmitters and receiver for DHU/DAU interface cable lines are National Semiconductor DM7830 and DM7820, respectively (or equivalent)

FIGURE 3-1. INTERFACE BETWEEN STATION COMPUTERS AND STATION ELECTRONICS DATA ACQUISITION UNIT (DAU)

Presence detector (PD), vehicle switch tone verification, and disparity bits from the collision avoidance system are contained in the DAU data. Each DAU transaction consists of a series of sequential input words. The number of words is determined by the number of bits required to implement a station. Figure 3-2 shows the DAU data interface timing requirements. The transaction begins with the DAU interrupting the computer(s) upon the expiration of an 8 pps clock with a pulse CYCLE REQUEST A. At this time the first input word is on the data lines. The software reads this data which results in a DAU END OF CYCLE to the DAU. Approximately 6 microseconds after the DATA TRANSMITTED pulse the DAU makes the next word available and generates another interrupt with CYCLE REQUEST A pluse. The same sequence of events occurs until all of the words of the transaction have been read. At this time, the DAU again issues the CYCLE REQUEST A pulse. There is no valid input data at this time. This last pulse is used as a signal to the software to reset the DAU to the first sequential world via the CSR1 pulse. The DAU is also reset at power up.



• In dual mode data transmitted at the user interface will not occur until both computer strings data transmitted occur or until a timeout condition is detected.

**Pulse width of the end of cycle (EOC) to the DHU is 400-ns to 2-μs wide.

••• Used to clear REQ A flip-flop after last valid data word. CSR1 gated to DAU interface as a DAU reset.

PD sequence occurs every 120 to 125 ms

FIGURE 3-2. DAU DATA INTERFACE TIMING REQUIREMENTS

Computer-SPE-DAU Interface Logic and Dual Computer Operation Requirements.

Because the DAU, which uses short pulses for control, must interface with the DR11C's in both computers which require an interrupt handshake logic the following interface shall be used. DAU data shall be made available to both computers while in dual mode of operation. The data shall be routed to the current primary computer only, during single computer mode. The DAU CYCLE REQUEST A shall be connected via FLIP/FLOPS to the DR11C REQUEST A lines. These FLIP/FLOPS shall be reset by raising DATA TRANSMITTED or the coincidence of DATA TRANSMITTED in dual mode. CSR1 from either computer shall also reset the associated FLIP/FLOP to allow acknowledgement of the last DAU CYCLE REQUEST A pulse without asking for more data. The DR11C CSR1 lines shall be wired such that the DAU reset pulse is generated by setting the primary computer CSR1 bit to a one. DR11C DATA TRANSMITTED shall be connected to the DAU END OF CYCLE so that the reading of data from the DR11C will transmit an acknowledge to the DAU. In dual mode, a timer shall be provided such that if one DATA TRANSMITTED pulse occurs and the coincident DATA TRANSMITTED pulse from the DR11C is more than 50 ms (+10 percent) late, the following actions are taken:

1. A DAU END OF CYCLE (DATA TRANSMITTED) shall be sent.

2. DR11C REQUEST B of the computer which sent DATA TRANSMITTED shall be set. (CSR0 from the same computer shall reset REQUEST B.)

3. The DAU CYCLE REQUEST A FLIP/FLOP for the computer which sent DATA TRANSMITTED shall be reset.

4. The REQUEST B FLIP/FLOP (ERROR) shall enable logic to allow subsequent DATA TRANSMITTED to be gated as if it were in single computer mode.

5. The error condition shall inhibit further timeouts until the REQUEST B FLIP/FLOP (ERROR) is cleared with CSR0.

If the interface is commanded from dual mode to single mode, the dual mode timer shall be inhibited. Also, the completion of any transaction which is waiting for an action from the removed companion computer shall be completed immediately.

SPE Mode Control Requirements. A SPE mode control register shall be provided in each computer. At the stations this register shall provide the capability to set the SPE logic to the following modes:

1. Set the selecting computer, be it A or B, to prime single mode.

2. Set the selecting computer to prime dual mode with the other computer as backup in dual mode.

Figure 3-3 shows the required and only state transitions for the station SPE in the form of a state transition diagram.

The central SPE mode control requirements are similar to the station except that only two modes are required as follows:

1. A computer outputs control the mimic and B outputs are ignored.

2. B computer outputs control the mimic and A outputs are ignored.

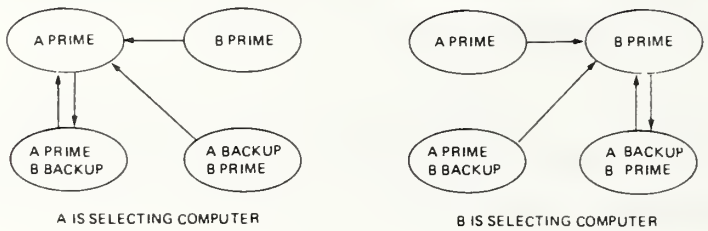


FIGURE 3-3. STATION SPE STATE TRANSITION REQUIREMENTS

3.1.3 Software Requirements

This section identifies the requirements levied on the software to provide the dual string redundancy for the redundant computing system.

The software is required to set central and station SPE to the proper modes for the various configurations. The prime string must set the central and stations SPEs to prime single mode upon system startup. When the backup is loaded the prime stations must set the station SPEs to dual mode. On switchovers and seize controls the new prime string must command the SPEs to prime single mode within 500 ms following a failure. The software is required to identify to the operator the string to which the central SPE is set and, thus, the string which is driving the mimic board.

The software is required to allow either string to be loaded as the prime string. The software is required to identify the prime and backup string to the operator and to maintain backup armed status. The definition of an armed backup is the state in which the backup string is available and prepared to automatically take over control of the system as a prime.

The software must allow the backup string to be gracefully brought on line at anytime with no interference to system operation. This includes loading the backup string and then synchronizing the backup string to the processing state of the prime string. Bringing the backup system on line provides some interesting problems. When the backup is initially loaded, it has no information about the state of the system. System inputs such as vehicle downlinks and vehicle PD hits must be ignored by the backup string until it is synchronized to the prime. Only after synchronization can the backup intelligently process system inputs. Once the backup is synchronized both strings must receive the same inputs in order to maintain synchronization. The software is required to pass the central operator CRT inputs from the prime to the backup and discard CRT inputs on the backup. Each central is

required to output to its own CRT screen. The software is required to provide the operator the capability to synchronize or to resynchronize the backup string with no interference to passenger service. The data synchronization must not require the backup to be disarmed for more than 10 seconds.

The software is required to monitor the computing system for failures so that the software can perform reconfigurations automatically. The computing system failures are divided into three categories: critical, non-critical, and pseudo-non-critical. Critical failures are failures which render the computer string unable or unreliable to control system operation. Critical failures require a reload of the computer string on which they occur. Critical failures include failures of a computer CPU, memory, and software, as well as failures of safetone output, modems, and system operator CRT.

Failures of non-essential peripheral devices are defined as non-critical failures. Non-critical failures must not interfere with system operation, and the failed device must be recovered without a reload of the computer string. Non-critical failures include failures of the magnetic tape units, mimic board, line printers, teletypes, disks, DSU, paper tape reader/punch, passenger boarding displays, and the string to string bus links. Pseudo-non-critical failures are almost non-critical failures. They include failures of the uplink, downlink, and data acquisition unit. These failures require guideway power removal in the control zone of the station in which they occur since the data these devices handle is safety critical. However, the software is required to provide pseudo-non-critical device recovery without a reload of the computer string.

The software is required to respond to all prime string critical failures by initiating an automatic switchover to the backup computing string if the backup is armed or by initiating a system wide guideway power shutdown if the backup is unarmed. The software must respond to all backup string critical failures by the prime seizing control in prime single mode and by removing the backup string from the system configuration. The software is required not to allow a computing string which has

detected a critical failure to become the prime string or to become an armed backup until the failed string has been reloaded. The software must respond to critical failures while in single string mode so as to continue operation in a degraded mode after shutdown of guideway power (i.e., to continue to monitor the system, track vehicles, update safe tones, update vehicle location data on disk, and allow any system operation within the physical limitations imposed by the failure except the reapplication of guideway power). This requirement makes system startup faster after a failure since the system is brought down in a graceful manner and is in a defined and ordered state.

The software is required to report to the system operator all equipment failures it detects except when the nature of the equipment failure prevents this. The software must notify the system operator of reconfigurations and identify which string has become the prime controlling string. The software is required to allow the operator to change the prime responsibility over to an armed backup string via a CRT keyboard command. The software must also allow diagnostics and troubleshooting to be performed on an unloaded backup system without affecting the system operation by the prime.

3.2 Detailed Design of the Redundant Computing System Hardware Elements

As can be seen from the discussion thus far the heart of the dual string redundancy scheme is the SPE logic which provides the control of the timing signals to maintain proper synchronization of the two computer interface units while the system is in dual mode of operation or is being changed from single to dual or vice-versa. This section describes the detailed design of the SPE equipment. To develop the design of SPE timeout error detection module a detailed timing diagram for the DAU interface is presented.

3.2.1 Data Acquisition Unit (DAU) Input SPE Equipment

The DAU data includes presence detector (PD), vehicle switch tone verification, and collision avoidance system disparity data. Figure 3-4

shows a block diagram of the DAU SPE design and the functional interface to the single thread DAU electronics and dual station computers.

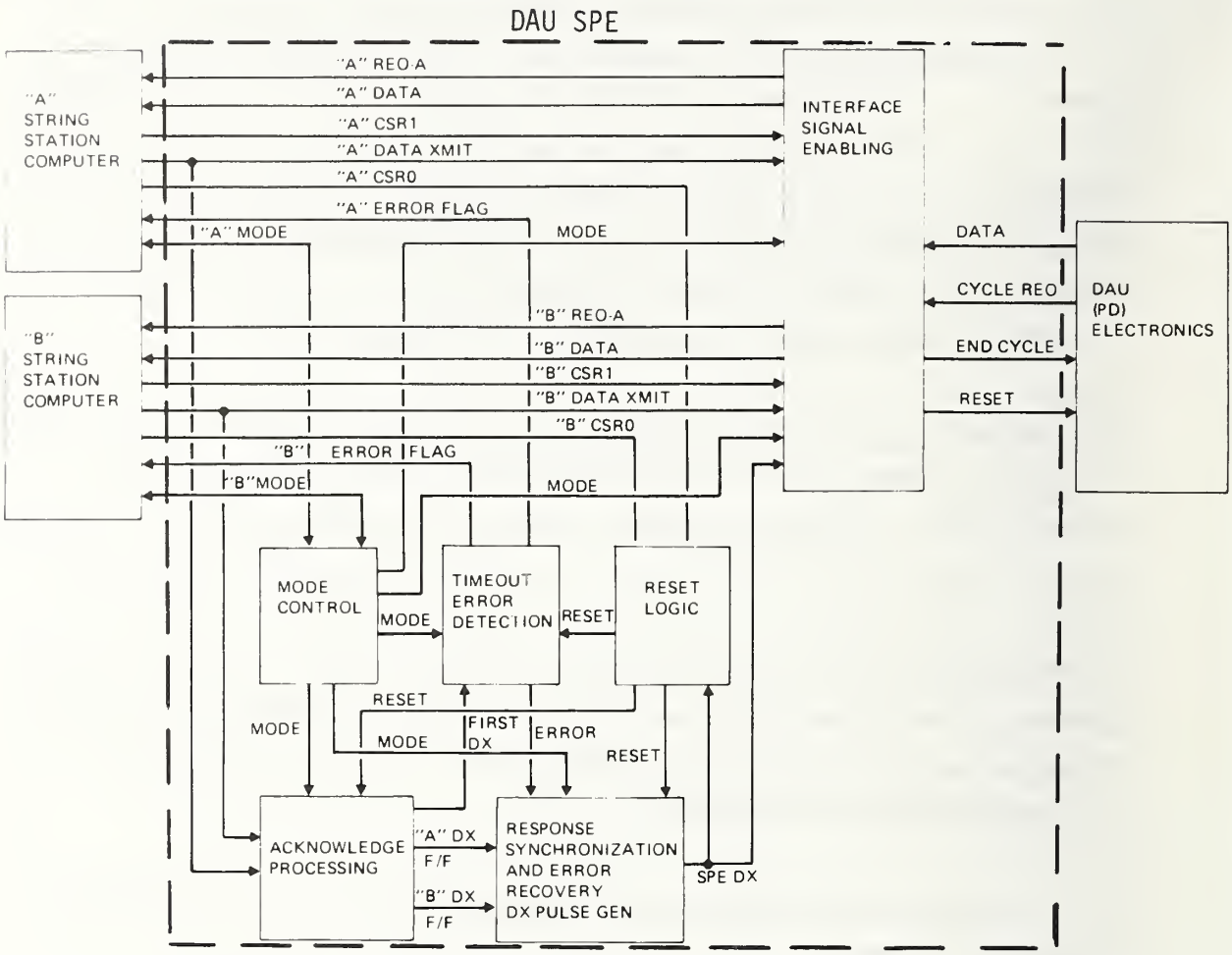
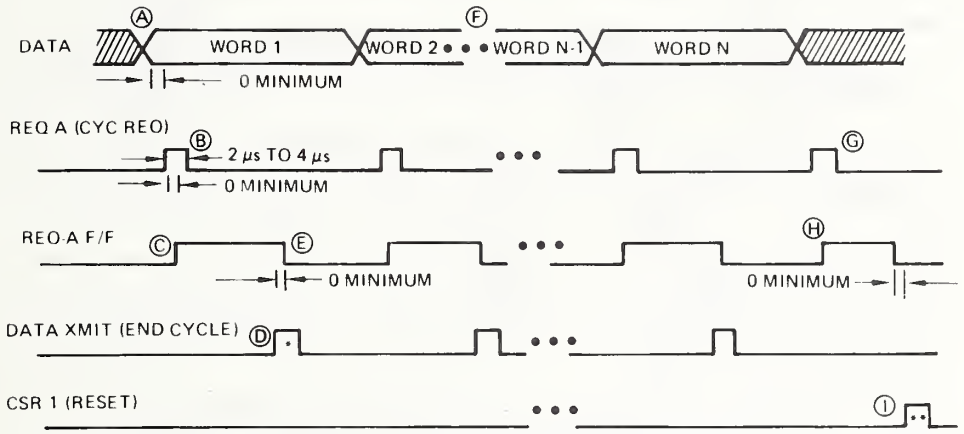


FIGURE 3-4. DAU SPE FUNCTIONAL BLOCK DIAGRAM

The timing at the DAU user device interface, which is the relationship for a single string system, is shown in Figure 3-5, and an explanation of the timing signals is given in Table 3-1. For dual string operation the sequence at the user interface is the same, but since the SPE must interface with two computers additional timing relationships to these interfaces is required. To show these relationships the dual mode control timing signals are added to the timing signals shown in Figure 3-5 and the composite is given in Figure 3-6 and in Table 3-2. On the composite the non-scripted bubbles represent the sequences which occur at the user interface and agree with those shown for the single string mode of operation. Once these timing interfaces were developed they were used to develop the control logic for dual computer operation.



*End cycle is 400 ns to 2 μs

**CSR1 is software-controlled with bit set/bit clear instructions.

FIGURE 3-5. DAU DATA SINGLE STRING TIMING

TABLE 3-1. DAU DATA SINGLE STRING TIMING

A) The DAU (user device) puts the first data word it wishes to transmit to the computer on the data lines.

B) After the data has settled and deskewed, a Req-A pulse is issued by the DAU.

C) The Req-A pulse directly sets a flip-flop to hold the request.

D) The computer will complete any processing at higher priorities then will read the data lines and issue a data xmit (end cycle) pulse to acknowledge the transaction.

E) The data xmit pulse clears the Req-A flip-flop and signals the user device that the computer is ready for word 2.

F) The above sequence, steps A through E, is repeated for each of the N words which comprise one data set.

G) An extra Req-A pulse is issued to signal the computer that the set is complete.

H) The Req-A pulse directly sets a flip-flop to hold the request.

I) The computer will complete any processing at higher priorities then will respond respond to the request by issuing a CSR1 (reset) pulse. The CSR1 pulse clears the Req-A flip-flop and signals the user device that the computer is ready for the next data set.

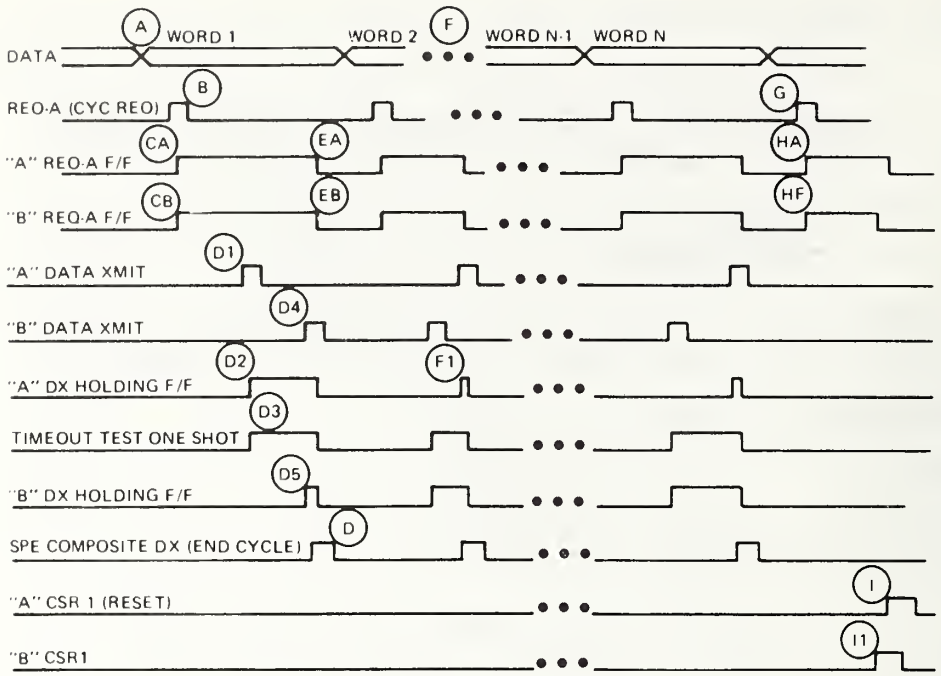


FIGURE 3-6. DAU DATA DUAL STRING TIMING (A STRING PRIME)

TABLE 3-2. DAU DATA DUAL STRING TIMING

A) The DAU (user device) puts the first data word it wishes to transmit to the computer on the data lines.

B) After the data has settled and time has been allowed for deskewing, a Req-A pulse is issued by the DAU.

CA) The Req-A pulse directly sets a flip-flop for each computer
CB) string to hold the request.

D1) One of the two computers will complete any processing at higher priorities first, then will read the data lines and issue a data transmit pulse to acknowledge the transaction.

D2) To assure the data will not change until the slower computer has responded, the data transmit pulse is not passed to the DAU (user device). Instead, it is used to set a DX holding flip-flop.

D3) So that the faster computer and the user will not be locked up waiting for the slower computer, either computer's data transmit will start a timeout test one shot. The monitor will normally be terminated by the end cycle pulse, but if the normal end cycle does not occur within the 50 ms timeout margin, a special sequence will occur to cause an end cycle pulse and to notify the good (faster) computer of the error. These sequences are shown in Figure 3-6.

TABLE 3-2. DAU DATA DUAL STRING TIMING (CONTINUED)

D4) The second (slower) computer will complete any processing at higher priorities, then will read the data lines and issue a data transmit pulse to acknowledge the transaction.

D5) The slower computer's data transmit in turn sets its associated DX holding flip-flop. This function is symmetrical so that it does not matter which is first.

D) As both DX holding flip-flops become set a SPE composite DX pulse is generated. This data transmit (end cycle) pulse is output to the DAU to acknowledge the transaction to indicate that the computers are ready for word 2.

EA) The SPE composite data transmit pulse (end cycle) clears both Req-A
EB) flip-flops allowing the computers to process word 2.

F) The above sequence, steps A through E, is repeated for each of the N words which comprise one data set.

F1) The order in which the internal logic sequences will vary as one computer or the other responds first. The signals described by the D-steps are shown first for "A" computer responding first and then for "B" computer responding first.

G) An extra Req-A pulse is issued by the DAU to signal the computers that the data set is complete.

HA) The Req-A pulse directly sets a flip-flop for each computer string
HB) to hold the request.

I) The prime string computer will complete any processing at higher priorities then will pulse the CSR1 line in response to the request. The CSR1 pulse from the prime string computer signals the user device that the computers are ready for the next data set and clears its own Req-A flip-flop.

IA) The backup string computer will complete any processing at higher priorities then will pulse the CSR1 line in response to the request. The CSR1 pulse from the backup string computer clears only its own Req-A flip-flop and is not gated to the user device.

First the functions which are required to support the normal error free sequence were determined. Then various failures were considered and the functions which are required to recover from the failure without interrupting the transaction with respect to the user device were added. This led to the design illustrated in the functional block diagram in Figure 3-4. The following presents the functions performed by each block in this figure and the reasons they are required.

Mode Control. The system has four basic modes of operation.

- "A" string computers in single string operation
- "B" string computers in single string operation
- Dual string operation with "A" String computer as prime
- Dual string operation with "B" string computer as prime

The mode control logic accepts mode control information from the SPE control interface, tests the cipher code to ensure that a legal change is being requested and then makes the change. The legal sequences are those previously shown in Figure 3-3, Station SPE State Transition Requirements. The mode control logic supplies the following signals to the rest of the SPE control logic:

"A" Single - "A" string enabled "B" disabled

"B" Single - "B" string enabled "A" disabled

"A" Prime - "A" string prime in dual or "A" in single mode

"B" Prime - "B" string prime in dual or "B" in single mode

"A" Enabled - Any mode except "B" single string

"B" Enabled - Any mode except "A" single string

"Dual" - Either string in dual mode

Interface Signal Enabling. The interface signal enabling logic provides the gating and holding logic so that the following data routing will be performed:

- Only the prime string data, whether the system is in single string or in dual string mode, is presented to the user device.

- Data from the user device is presented to both computers when in dual mode. This data will remain valid until both computers have acknowledged reading it.

- Service requests are presented only to active computers when in single string and are presented to both computers when in dual mode.

- Acknowledge signals are routed from the active computer when in single string and from the SPE response syncronization pulse generator when in dual mode. This is the technique used to provide valid data until both computers have read the data.

- User reset signals are routed from the prime string computer only.

Acknowledge Processing. The acknowledge processing provides storage (holding) pulses and signals the timeout monitor as each cycle is started.

Timeout Error Detection. The timeout error detection logic performs the function of assuring that both computer strings are syncronized to within 50 ms of each other. If they are not, an error will be reported to the computer which has completed its part of the transaction which failed. To accomplish the monitoring task the following functions are provided:

- The timeout monitor is enabled and an error will be reported only when the SPE is in dual mode.

- The "A" string error flag will be set and a SPE device timeout reported to the "A" string when the "A" string computer has acknowledged a request and the "B" string computer has failed to do so within 50 ms.

- The "B" string error flag will be set and a SPE device timeout reported to the "B" string when the "B" string computer has acknowledged a request and the "A" string computer has failed to do so within 50 ms.

- Further error reporting is disabled when the first error occurs until the error flag is reset.

- A reset error pulse (CSRO) from the "A" string computer will clear the "A" string error flag.

- A reset error pulse (CSRO) from the "B" string computer will clear the "B" string error flag.

Response Syncronization and Error Recovery. When in dual mode of operation the data transmit pulse, which is the acknowledge signal to the user device, is generated by the SPE logic. Since this pulse completes each transfer and signals the user that the next transfer can start, this pulse must occur or the system will lock-up. Because of this it is necessary to generate the acknowledge pulse at the completion of both normal cycles and for cycles in which a reconfiguration action to recovery from an error has taken place. To maintain non-interrupted data flow from the DAU interface to the redundant computer system the acknowledge signal (SPE DX) is generated by the responce syncronization and error recover module for the following cases:

- When the "A" string PD data xmit holding flip-flop and the "B" string PD data xmit holding flip-flop both are set before a timeout error occurs. This is the normal path and provides string syncronization by waiting until both computers respond before completing an input cycle.

- When the timeout error from either "A" or "B" string is detected. This case is required to recover either from a failure of the DAU interface controller in one of the computers or from an internal computer failure which results in the controller no longer being serviced.

- When an "A" string data xmit pulse occurs with the "A" string error flag (indicating "B" string failure) set. This provides for interface operation for the period of time between a timeout error until SPE mode reconfiguration is accomplished.

- When a "B" string data xmit pulse occurs with the "B" string error flag (indicating "A" string failure) set. This provides for interface operation for the period of time between a timeout error until SPE mode reconfiguration is accomplished.

- When the "A" string PD data xmit holding flip-flop is set and the SPE mode is changed to "A" single string. This provides for the completion of a cycle for the case in which a failure which does not involve this particular interface has occurred. The "B" would have responded but couldn't because it was switched off. If this initiate function were not provided, the interface would hang-up.

- When the "B" string PD data xmit holding flip-flop is set and the SPE mode is changed to "B" single string. This is the mirror of the above except that the "B" string is assuming control.

Reset Logic. The reset logic clears all the storage (holding) logic at the completion of each cycle. The reset logic clears the appropriate error flags when the reset error pulses (CSRO) are received.

General SPE Design. In addition to the specific functions performed by the various blocks of the SPE logic, some general design requirements were also satisfied. The functions in the SPE are performed with respect to the "A" string or the "B" string, but it is a requirement that this

be transparent to the software (the computer interface). To accomplish this all computer interface register formats were designed so that the same masks, bit assignments,and instructions could be used in both computers to accomplish a function. The SPE logic then qualifies these interface signals by knowing which interface issued the command. This feature allows identical software to be used in both computer strings and subsequently allows the "A" and "B" computers in any given station to be interchanged when making up a computer string. Another feature which was designed into the SPE logic was the segmentation of the "A" string peculiar logic, the "B" string peculiar logic, and the remaining logic to the board level so that repair by replacement can be accomplished with minimal operational impact. A maintenance panel function was provided for each interface so that test configurations and equipment isolation could be provided by simply moving front panel connectors.

3.2.2 Destination Selection Unit and Vehicle Downlink Input SPE Equipment

As has been mentioned earlier, the station SPE equipment consists of three basic types. The destination selection unit and vehicle downlink SPE equipment is simular to that just described for the DAU inputs. These interfaces all provide data input to the computers which must be syncronized.

3.2.3 Vehicle Uplink, Collision Avoidance, and Mimic Output SPE Equipment

The second type of SPE equipment is used for vehicle uplink and collision avoidance system outputs at the stations and for mimic outputs at central. This type of SPE equipment is shown in a functional block diagram in Figure 3-7 for the vehicle uplink device. This interface, which outputs data from the computers, ties only the interface for the string which is selected as prime to the user device. Either the computer interface for the backup string is routed through internal SPE logic to simulate the control signals, if this is required, or the backup string interface is not used at all if the device does not use control lines.

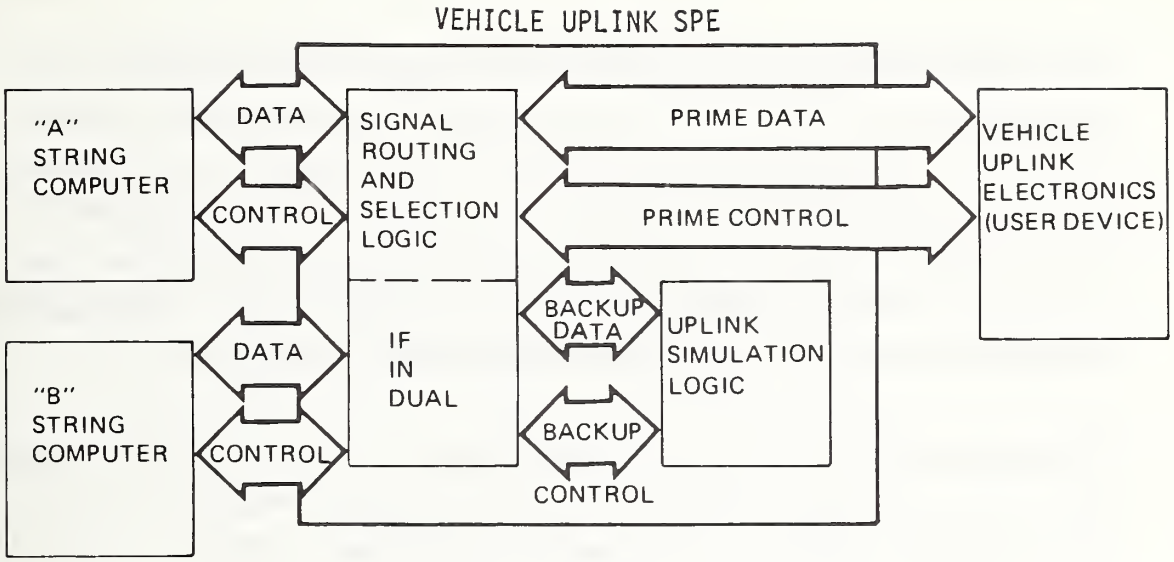


FIGURE 3-7. VEHICLE UPLINK SPE FUNCTIONAL BLOCK DIAGRAM

3.2.4 Passenger Boarding Displays and Electrification Trip SPE Equipment

The third type of SPE equipment is used for the passenger boarding displays in the stations and the electrification trip device in central. This SPE is simply a wire "or" of the two computers outputs. Isolation is provided by removing a front panel connector in the SPE maintenance panel.

3.2.5 Modem Reconfiguration Unit

Early trade studies which were conducted to determine whether the MPM redundant computing system should be capable of being reconfigured on a station by station basis or as a total string weighed in favor of the string approach. The primary factor was the added complexity to develop a system which could be reconfigured on a piece-wise basis. During the early testing and operational phase it became apparent that the symmetry of the system allowed for reconfiguration of the station computers on a station by station basis by simply swapping cables. This was desirable since computer failures did occur in the single operating string while the other string was being repaired. To accomplish this recabling in a rapid and controlled manner the modem reconfiguration

unit shown functionally in Figure 3-8 was added. When a station is reconfigured, the "A" and "B" stations are exchanged so that instead of central A being connected to station A and central B being connected to station B the role is reversed so that central A is connected to station B and central B is connected to station A. At the same time the remote bootstrap pushbuttons are switched so that the same operator procedure can always be used to load the prime or backup.

The modem reconfiguration unit is a manually controlled switching device which does not support automatic reconfigurations. However, it does allow a "good" string to be assembled and reloaded quickly.

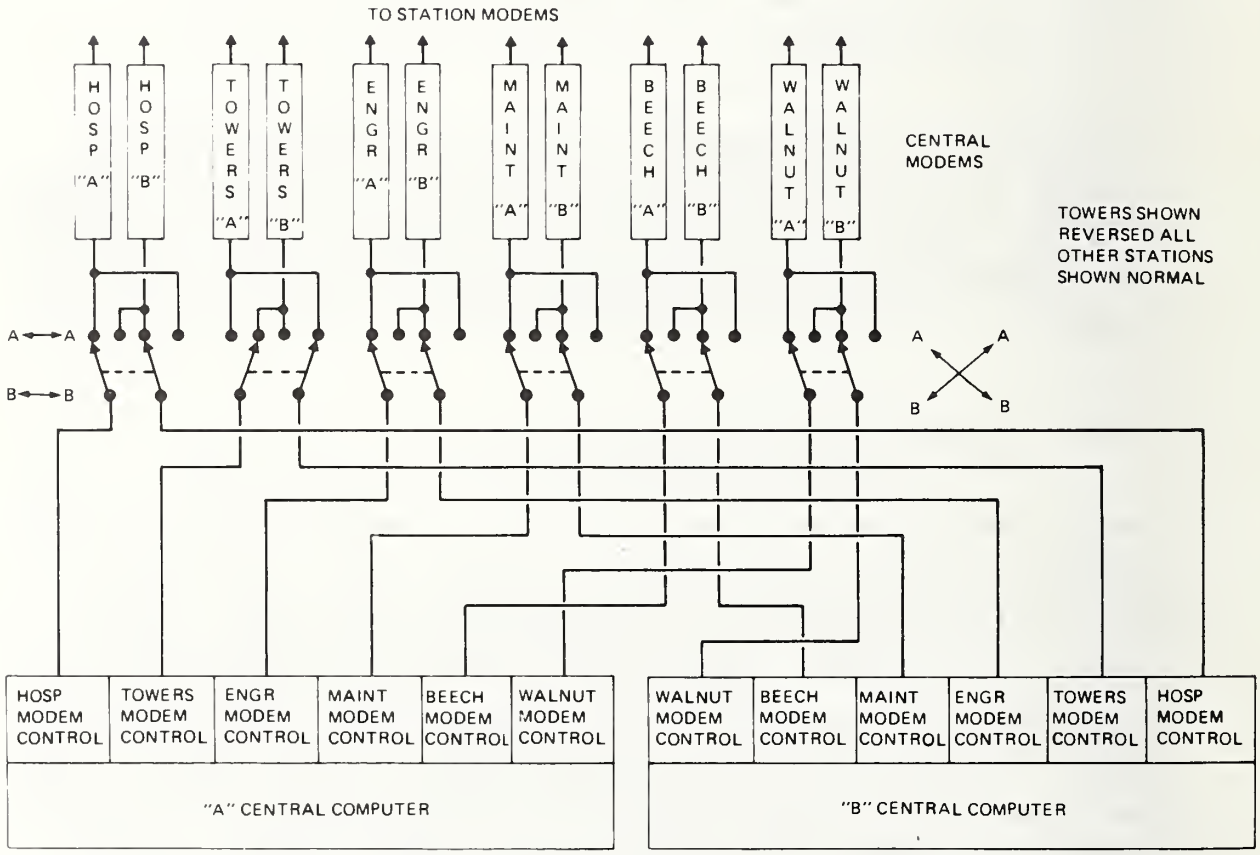


FIGURE 3-8. MODEM RECONFIGURATION UNIT FUNCTIONAL DIAGRAM

3.3 Detailed Design of the Redundant Computing System Software Elements

This section describes the detailed design of the redundant computing system software elements and shows how the design satisfies the requirements given in Section 3.1. As explained previously, the redundant functions are allocated to the executive software. The fact that a second string exists is transparent to the applications software except for display purposes and for implementation of dual string management operator CRT key actions, such as arm, synchronize, or changeover to the backup string. These key actions and the generation of certain displays precipitate calls to the executive to perform the redundant computing system functions. For these reasons, only the executive software is described here.

Figure 3-9 shows the executive program organization by module groups and modules. The set of routines listed is the subset of routines which provide the majority of the redundancy support. Other executive routines either have no roll in the redundancy functions or are only slightly impacted to support functions performed by the listed routines. The routines listed are not totally dedicated to the redundant functions. For example, the central loader determines prime or backup status on startup, but its major function is to load the central computer. From a routine count standpoint, the routines listed constitute approximately 20 percent of the total executive routines. The redundancy functions of the listed routines constitute approximately 2600 lines of executable code which is 20 percent of the executive. Including code and data the redundancy functions constitute 5 percent of the total operational software. The following sections describe the detailed design of the redundancy functions of the listed routines.

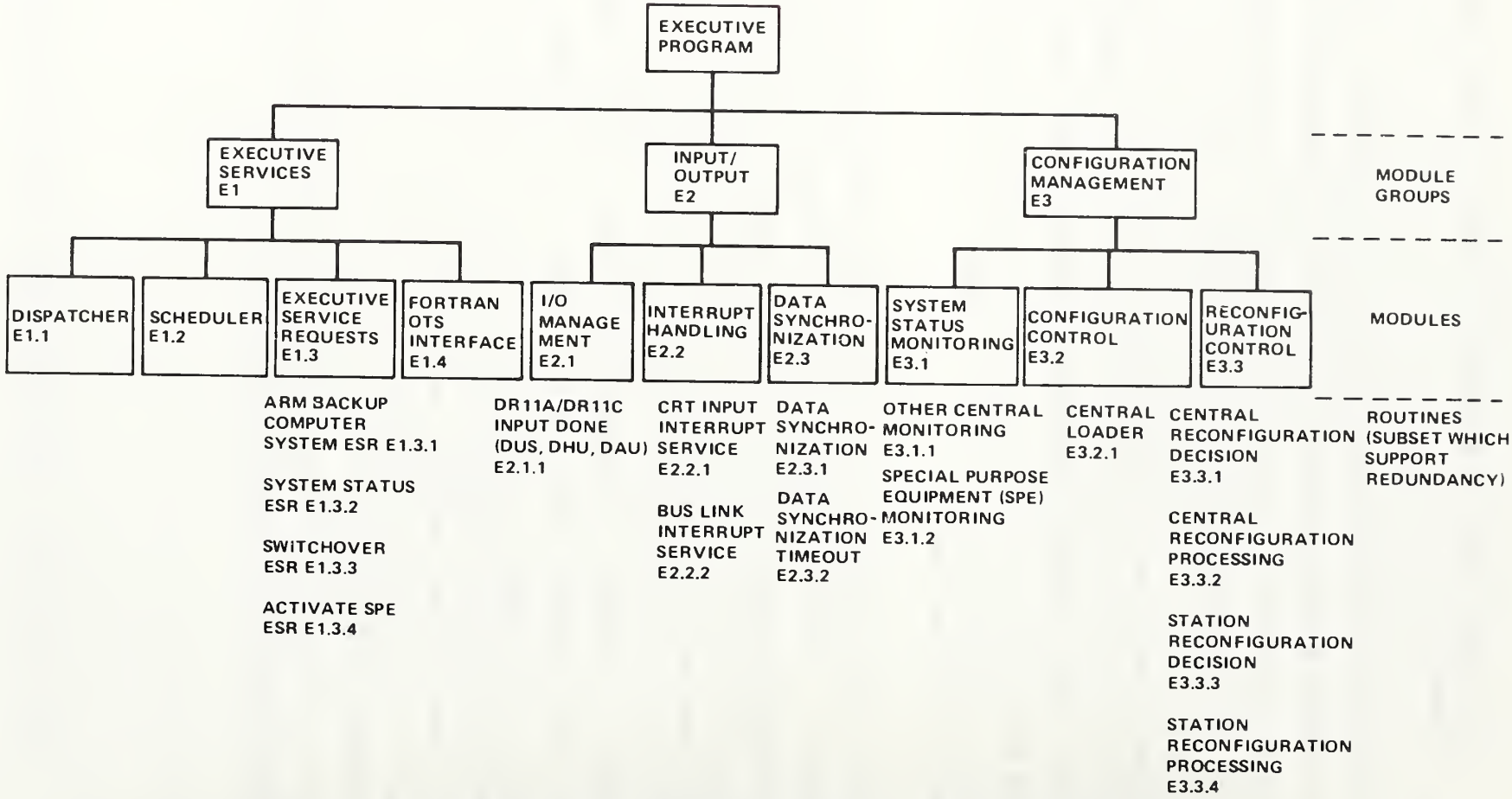


FIGURE 3- 9 EXECUTIVE ROUTINES WHICH SUPPORT REDUNDANCY WITHIN EXECUTIVE PROGRAM ORGANIZATION

3.3.1 Executive Service Requests

This section describes the detailed design of the redundancy functions of the routines in the Executive Service Request module. Executive Service Requests (ESRs) are called by the applications and executive to initiate or perform functions allocated to the executive such as device input/output. The following describes the redundancy functions of the ESRs.

Arm Backup Computer System ESR. The Arm Backup Computer System ESR provides CAP the capability to request arming or disarming the backup system. Until the backup is armed it is not elegible to take control automatically in the event of a prime string failure. The Arm Backup ESR exists only at central, and only central knows if the backup is armed or disarmed.

The Arm Backup ESR satisfies the requirement to maintain the backup armed status by arming and disarming the backup in response to operator commands. The armed/disarmed status is also modified automatically by Central Reconfiguration Decision on the basis of critical failures. The arm backup ESR prohibits an unprepared backup from becoming armed since central-to-central communications must be operating and since the backup must be synchronized before it can be armed.

Figure 3-10 shows the functional flow of the Arm Backup ESR. As can be seen from this flow, only the backup sets the backup to armed and only after data synchronization. However, either prime or backup can set the backup to disarmed.

Figure 3-11 shows a control/data flow for the Arm Backup ESR for the case of arming the backup. As can be seen from this flow this ESR is called by CAP in the prime and backup as a result of the arm or disarm backup operator CRT keyboard action on the prime keyboard. As explained previously, after data synchronization the prime CRT input characters are passed to both the prime and backup applications. The backup sets the backup to armed in the backup data base. The prime learns of the backup armed status by the backup status in the system

status table being periodically written to the prime memory as part of the other central monitoring function.

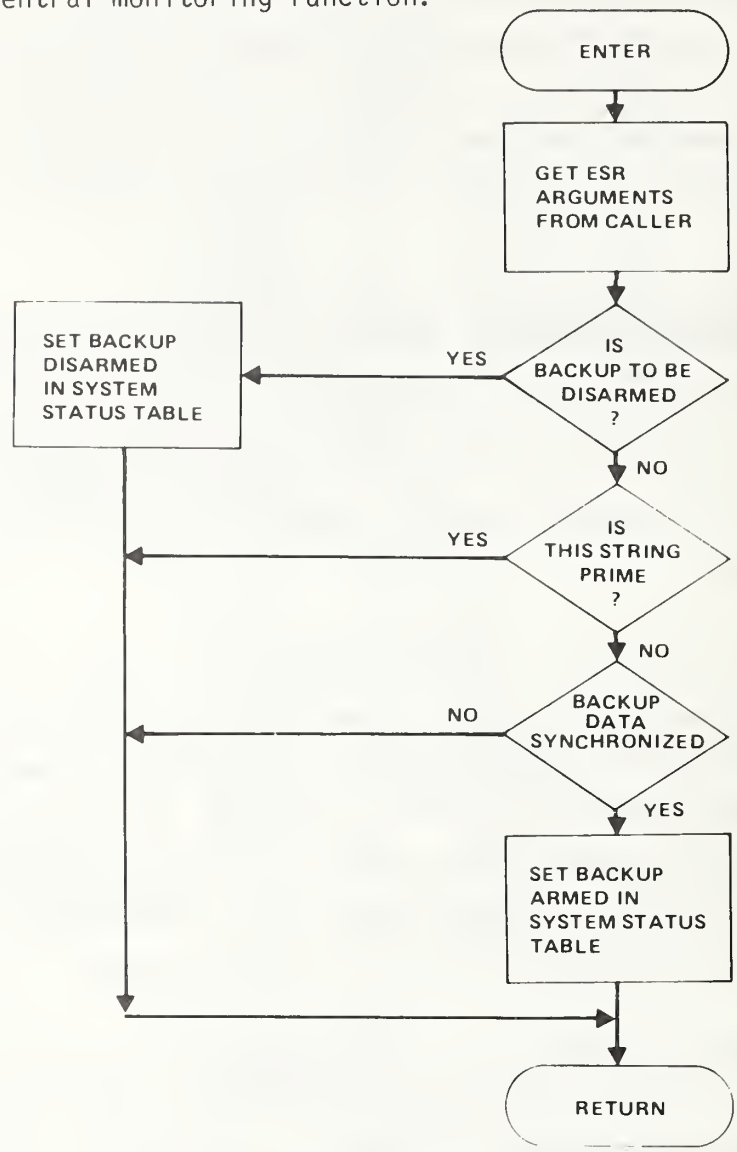


FIGURE 3-10. ARM BACKUP COMPUTER SYSTEM ESR

System Status ESR. The System Status ESR provides CAP the capability to determine prime/backup status, arm/disarm status, and central SPE status; in addition, it permits CAP to determine which computers in the network are loaded and running. Upon operator request, CAP displays this information on the CRT display. The System Status ESR exists at central only. Thus, CAP is the only application software which has access to this information. The prime/backup status is maintained in all computers by the executive software.

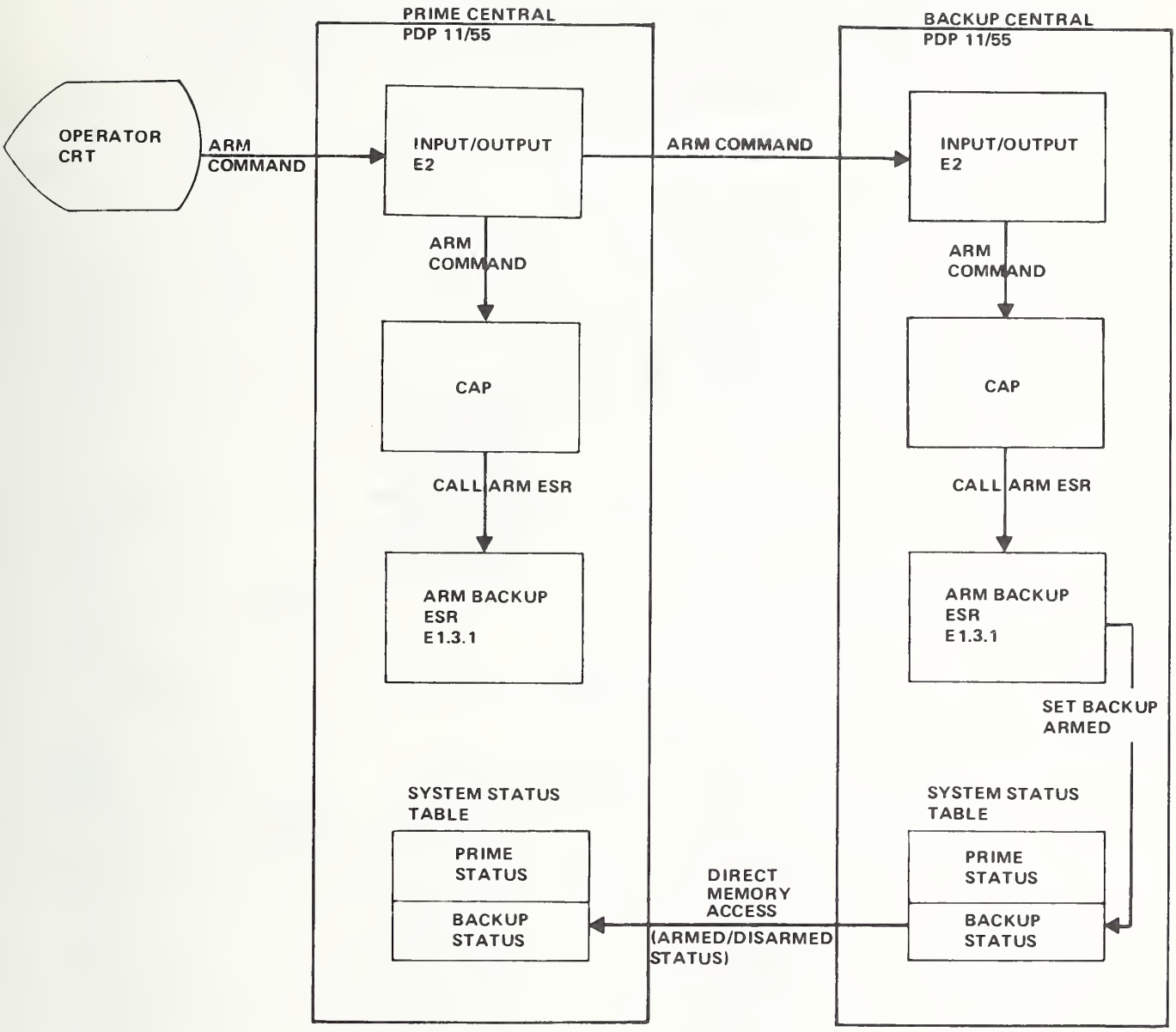


FIGURE 3-11. ARM BACKUP CONTROL/DATA FLOW

The System Status ESR satisfies the requirement to identify the prime/backup, armed/disarmed, and central SPE status to the operator by supplying this status to CAP upon request. CAP in turn displays the status to the operator on the CRT screen.

Figure 3-12 shows the functional flow of the System Status ESR. The System Status ESR determines the prime/backup status of the computer it is resident in and moves the contents of the system status table to the proper sections of the requested storage array. The ESR also

accesses the central SPE to determine the string to which the SPE is set and, thus, the string which is driving the mimic board. The ESR passes this status to the requestor.

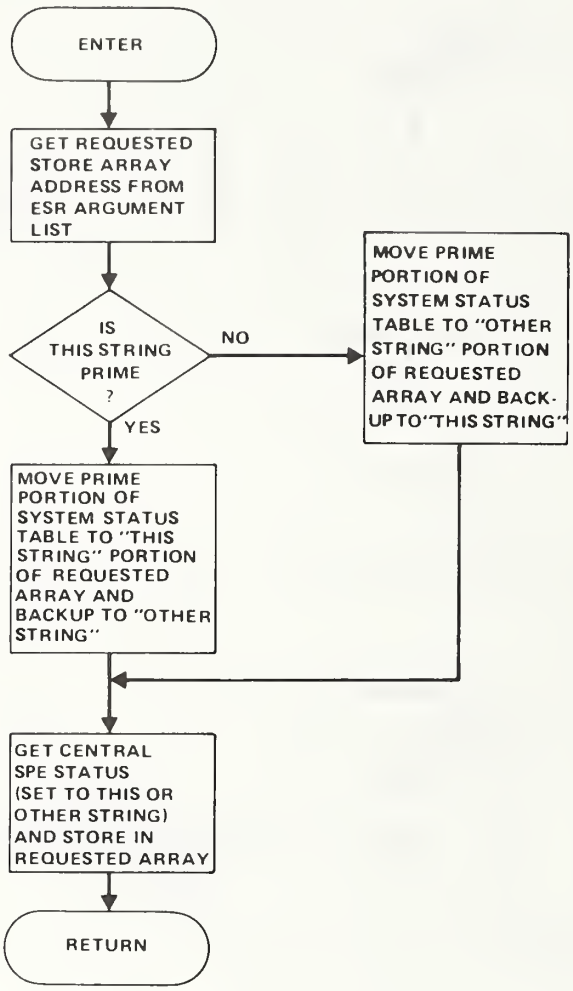

ENTER
GET REQUESTED STORE ARRAY ADDRESS FROM ESR ARGUMENT LIST
IS THIS STRING PRIME ?
NO
YES
MOVE PRIME PORTION OF SYSTEM STATUS TABLE TO "OTHER STRING" PORTION OF REQUESTED ARRAY AND BACK-UP TO "THIS STRING"
MOVE PRIME PORTION OF SYSTEM STATUS TABLE TO "THIS STRING" PORTION OF REQUESTED ARRAY AND BACKUP TO "OTHER STRING"
GET CENTRAL SPE STATUS (SET TO THIS OR OTHER STRING) AND STORE IN REQUESTED ARRAY
RETURN

FIGURE 3-12. SYSTEM STATUS ESR

Figure 3-13 shows a control/data flow for the System Status ESR. This flow shows the case in which data synchronization has taken place and prime CRT inputs are being passed to the backup. The system status is requested by the operator on the prime and the ESR is called in both the prime and the backup. The ESR accesses the system status table which is divided into prime and backup status sections. The prime system moves the prime portion to the "this-string" area of the requested array and the backup portion to the "other-string" area. The backup system reverses this procedure, moving the prime portion to the "other-string" area and the backup to the "this-string" area.

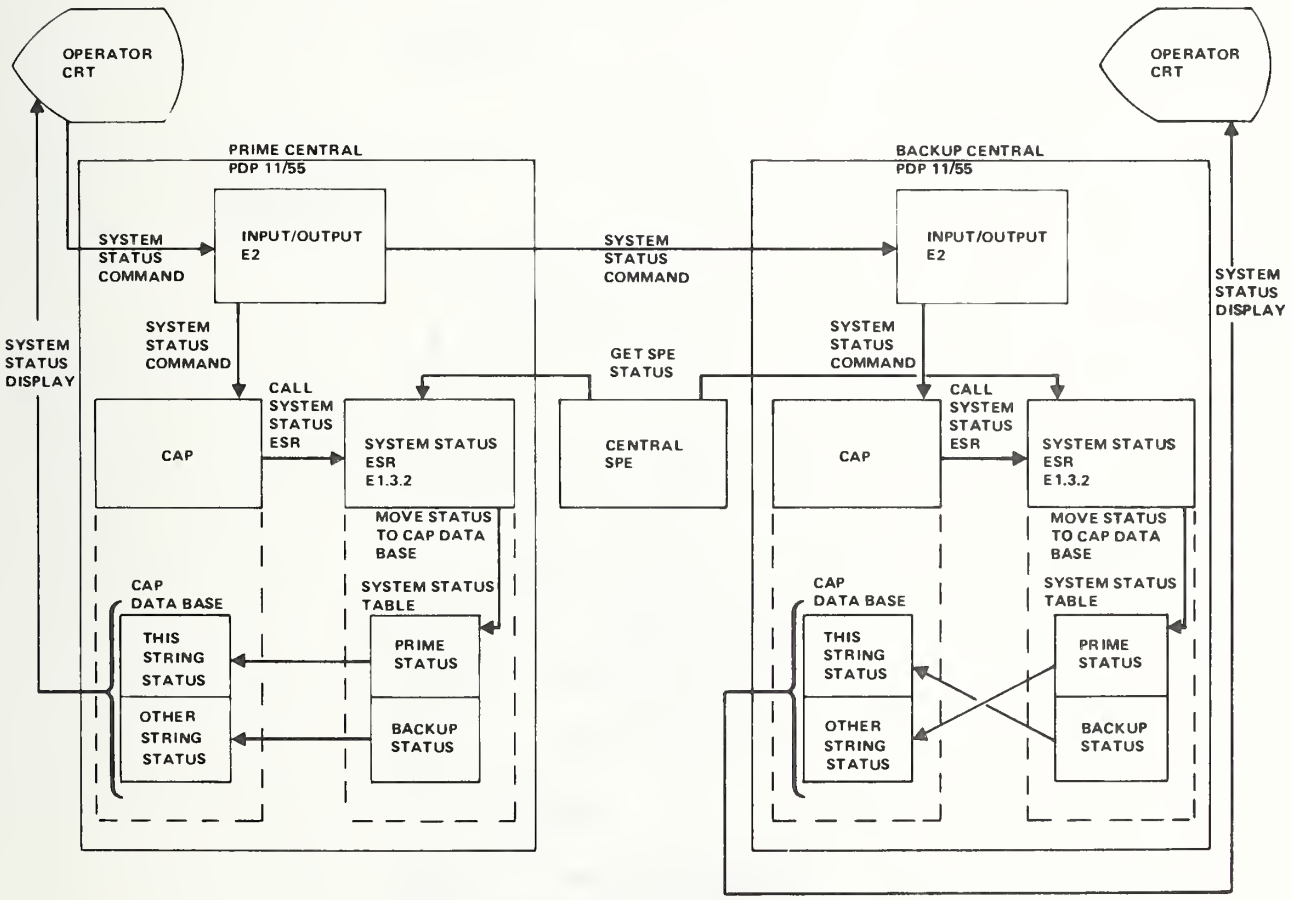


FIGURE 3-13. SYSTEM STATUS CONTROL/DATA FLOW

Switchover ESR. The Switchover ESR provides CAP the capability to request a switchover to the backup system. This enables the operator to make the backup system become the prime system. The Switchover ESR exists at central only.

The Switchover ESR satisfies the requirement to enable the system operator to change the prime responsibility over to an armed backup string via a CRT keyboard command by calling Reconfiguration Control to perform the actual switchover.

Figure 3-14 shows the functional flow of the Switchover ESR.

As can be seen from this flow, the ESR is ignored on the backup system and is only carried out on the prime if the backup is armed. This ESR performs a complex service in a very simple manner. It reports a critical failure of the prime string to the Reconfiguration Control module. Reconfiguration Control acts on the failure as it would on a genuine failure in the prime string by causing a switchover to the armed backup string.

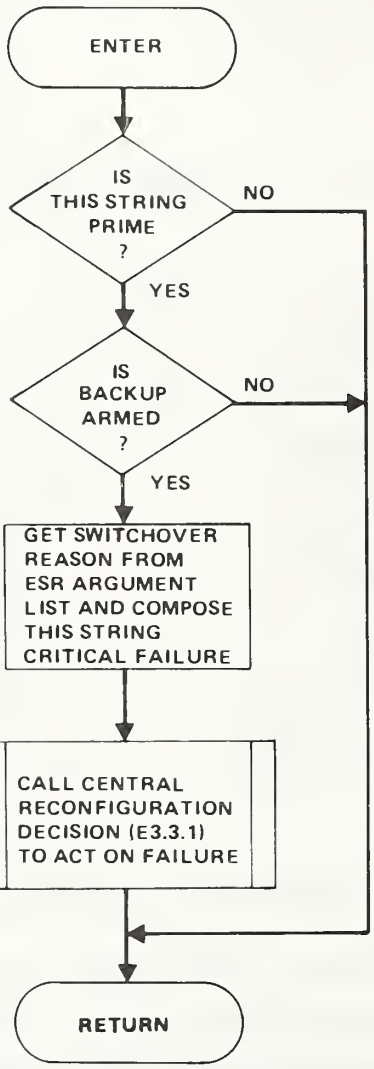
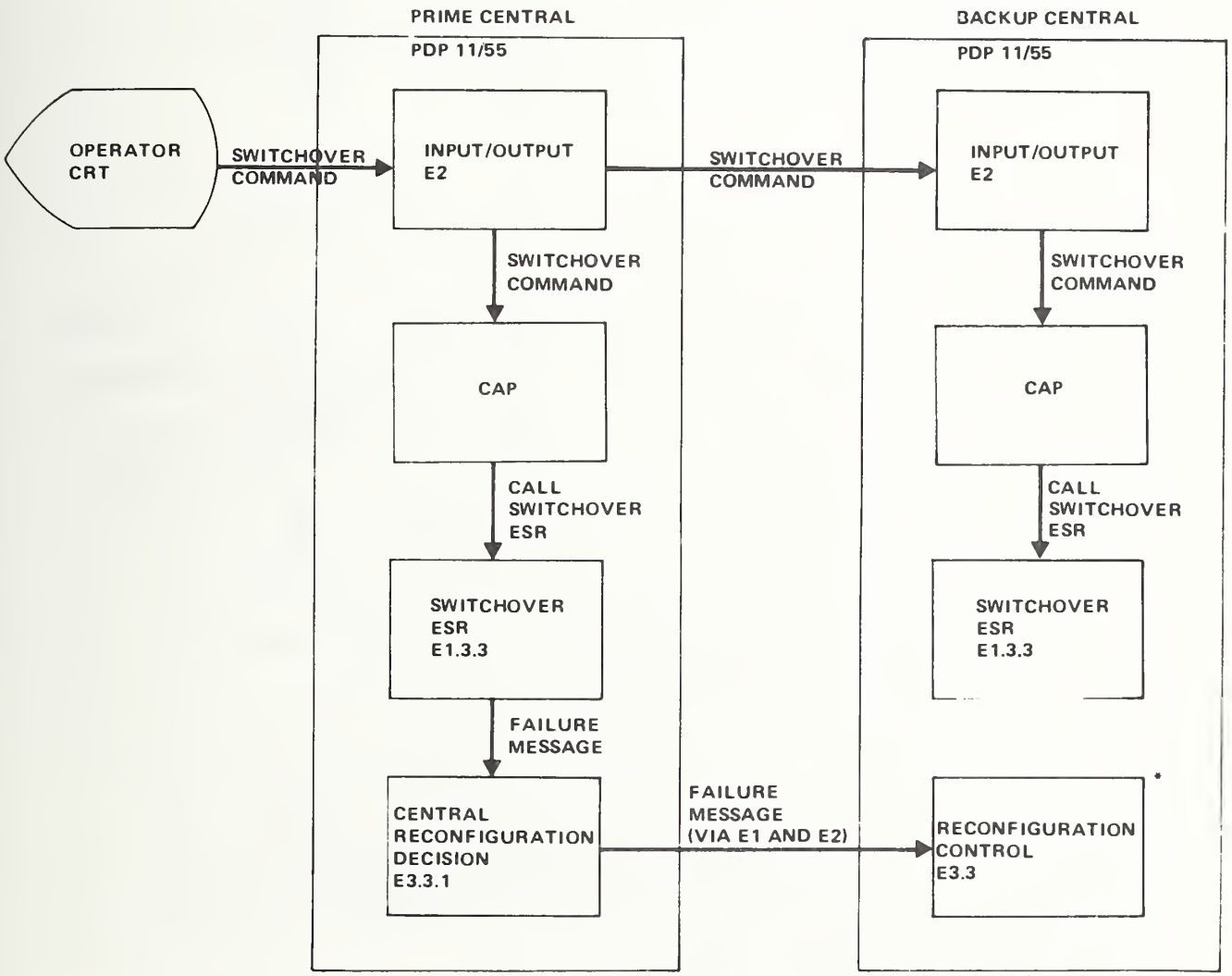


FIGURE 3-14. SWITCHOVER ESR

Figure 3-15 shows a control/data flow for the Switchover ESR. As can be seen from this figure, the operator requests a switchover to the backup on the prime CRT. The switchover key strokes are passed to the backup, and CAP calls the Switchover ESR in both the prime and the backup. Switchover ESR in the backup ignores the call. Switchover ESR in the backup ignores the call. Switchover ESR in the prime creates a "this-string" critical failure message and passes it to Central Reconfiguration Decision. Central Reconfiguration Decision passes the failure to the backup Reconfiguration Control. The backup Reconfiguration Control module believes that the prime string has failed and, therefore, seizes control in prime single mode thus effecting the desired switchover.



*Reconfiguration control receives the prime string failure message and effects a switchover to the armed backup in the same manner as for other prime string critical failures.

FIGURE 3-15. SWITCHOVER ESR CONTROL/DATA FLOW

Activate SPE ESR. The Activate SPE ESR sets the central and station SPEs to prime single mode. The Activate SPE ESR exists at central and at the stations.

The Activate SPE ESR satisfies part of the requirement to set SPE to the proper mode by setting the SPEs to prime single mode upon system startup. The rest of the requirement to set SPE to the proper modes on backup loading and on reconfiguration is performed by routines in System Status Monitoring and Reconfiguration Control.

Figure 3-16 shows the functional flow of the Activate SPE ESR. As can be seen from this flow, the ESR is ignored on the backup system. The ESR is also ignored at station if the station is already in dual mode. This ESR is called after the prime system is loaded, before any vehicle is moved, and before the backup string is loaded.

Figure 3-17 shows a control/data flow for the Activate SPE ESR. The ESR is called as a result of the operator CRT activate I/O command. The setting of SPE is tied to an operator command so that the operator controls when the SPEs are set to prime single mode. In certain situations this allows the operator to avoid creating a CAS disparity which requires a trip to the station to clear. As can be seen from Figure 3-17, the operator activate I/O command is passed to CAP which calls the Activate SPE ESR at central and broadcasts an activate message to the stations via the executive ESR and I/O services. The Activate SPE ESR in the prime central sets the central SPE to prime single mode which causes the prime and the prime only to control the central mimic board. The Station Application Programs (SAP) in each of the stations calls the Activate SPE ESR as a result of the activate message from CAP. The Activate SPE ESR in the stations sets the station SPEs to prime single mode and enables input interrupts for input devices on the SPE. The enabling of inputs for the devices on the SPE is performed after the setting of SPE so that inputs will not cause SPE timeouts if the SPE was previously left in dual mode.

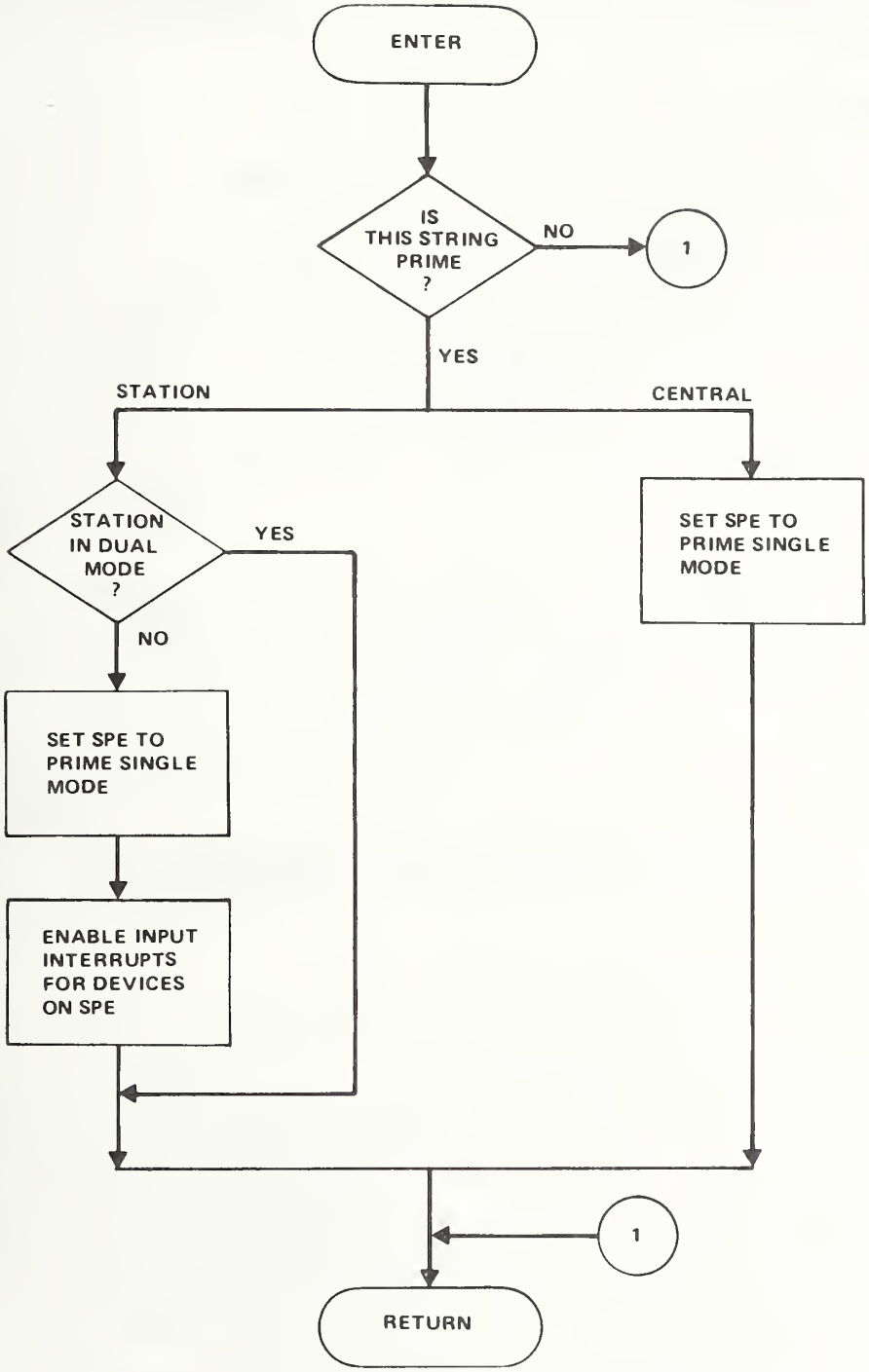


FIGURE 3-16. ACTIVATE SPE ESR

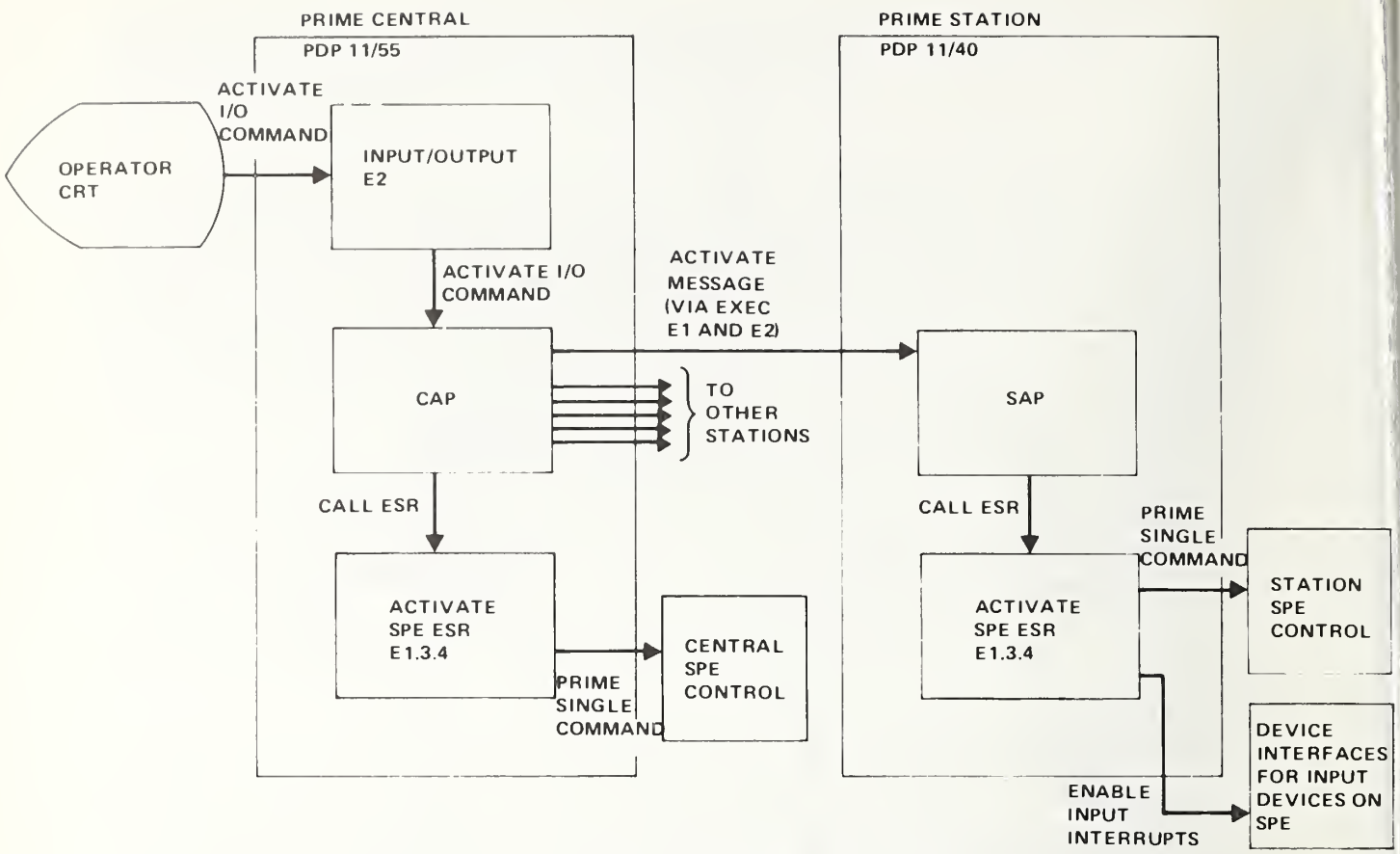


FIGURE 3-17. ACTIVATE SPE ESR CONTROL/DATA FLOW

3.3.2 I/O Management and Interrupt Handling

This section describes the detailed design of the redundancy functions of the routines in the I/O Management and Interrupt Handling modules. The I/O Management module performs the initiation and termination of input and output processing for the various computer interface devices. The initiation of I/O is performed by the I/O Initiation routine which calls a start routine for the specific device to be started. The termination of I/O is performed by the I/O Complete routine which calls a done routine for the specific device. The Interrupt Handling module performs processing between the start and completion of I/O. The Interrupt Handling module processes device interrupts, performs the processing necessary to continue device data transfer, and notifies I/O Management when the data transfer is complete. Most of the I/O device control routines are unaffected by the redundancy funtions.

DR11A/DR11C Input Done. The DR11A/DR11C Input Done routine queues the proper SAP task to process DHU, DAU, or DSU inputs. These devices and, thus, the routine exists only at the stations.

The DR11A/DR11C Input Done routine satisfies the requirement to allow the backup to be brought on line in a graceful manner. It does this by passing the DHU, DAU and DSU inputs to SAP on the backup only after the backup has been data synchronized. This prevents the backup applications from attempting to process system inputs before they know the current state of the system. Processing the inputs by the backup before data synchronization would cause a rash of anomalies and confusion in the backup system.

Figure 3-18 shows the functional flow of the DR11A/DR11C Input Done routine. As can be seen from this flow, the system inputs are always passed to the applications on the prime, but only after data synchronization are they so passed on the backup. Prior to data synchronization on the backup, the DR11A/DR11C Input Done routine returns the executive buffers which contain the input data to the available space list. This cleans up the executive data base after the input and effectively ignores the system inputs.

Figure 3-19 shows a control/data flow for the DR11A/DR11C Input Done routine. The station SPE splits the system inputs and presents them to the prime and the backup simultaneously. The DR11A/DR11C Input Interrupt routine receives the input interrupts and buffers up the input data. On completion of a message the DR11A/DR11C Input Done routine is called. The ignoring of the inputs is performed in the done rather than in the interrupt routine because the former provides a cleaner point to make the break.

CRT Input Interrupt Service and Bus Link Interrupt Service. The CRT Input Interrupt Service routine responds to the operator CRT input interrupts queueing the proper applications task to process the input character. The Bus Link Interrupt Service routine responds to the bus link interrupts to provide the communications between computers at the same location in the opposite string.

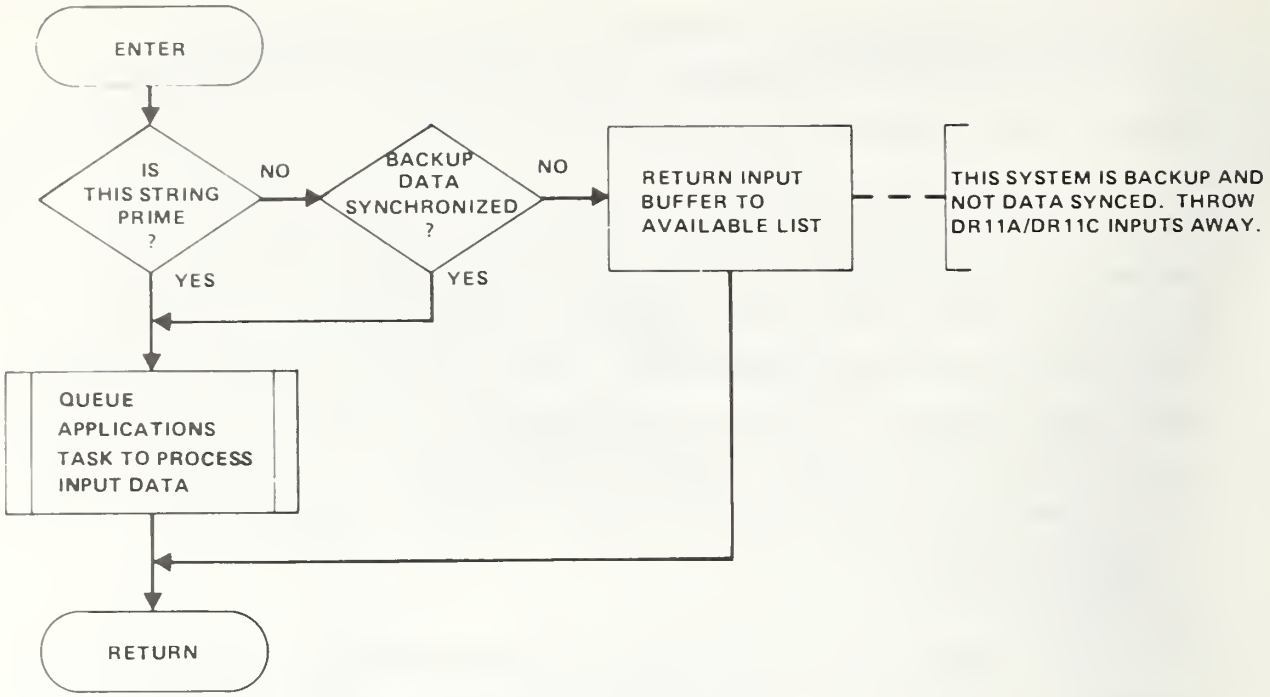


FIGURE 3-18. DR11A/DR11C INPUT DONE (DSU, DHU, DAU)

Together these two routines satisfy the requirement to pass the central operator CRT inputs from the prime to the backup and to discard CRT inputs on the backup after data synchronizaion. The purpose of passing the inputs to the backup is to provide the same inputs to both strings so that they will stay in synchronization.

Figure 3-20 shows the functional flow of the CRT Input Interrupt Service routine. As can be seen from this flow, after the backup has been data synchronized the CRT inputs are passed to the applications on the prime but are ignored on the backup. Also, after data synchronization, the prime outputs the CRT input characters to the backup via the bus link. As can be seen from Figure 3-21 the Bus Link Interrupt Service routine passes the CRT input characters received from the prime to the CRT Input Interrupt Service routine on the backup. Notice in Figure 3-20 that after a critical error occurs, the prime CRT inputs are no longer passed to the backup. The strings become independent so that each string can be queried to determine the nature of the failure.

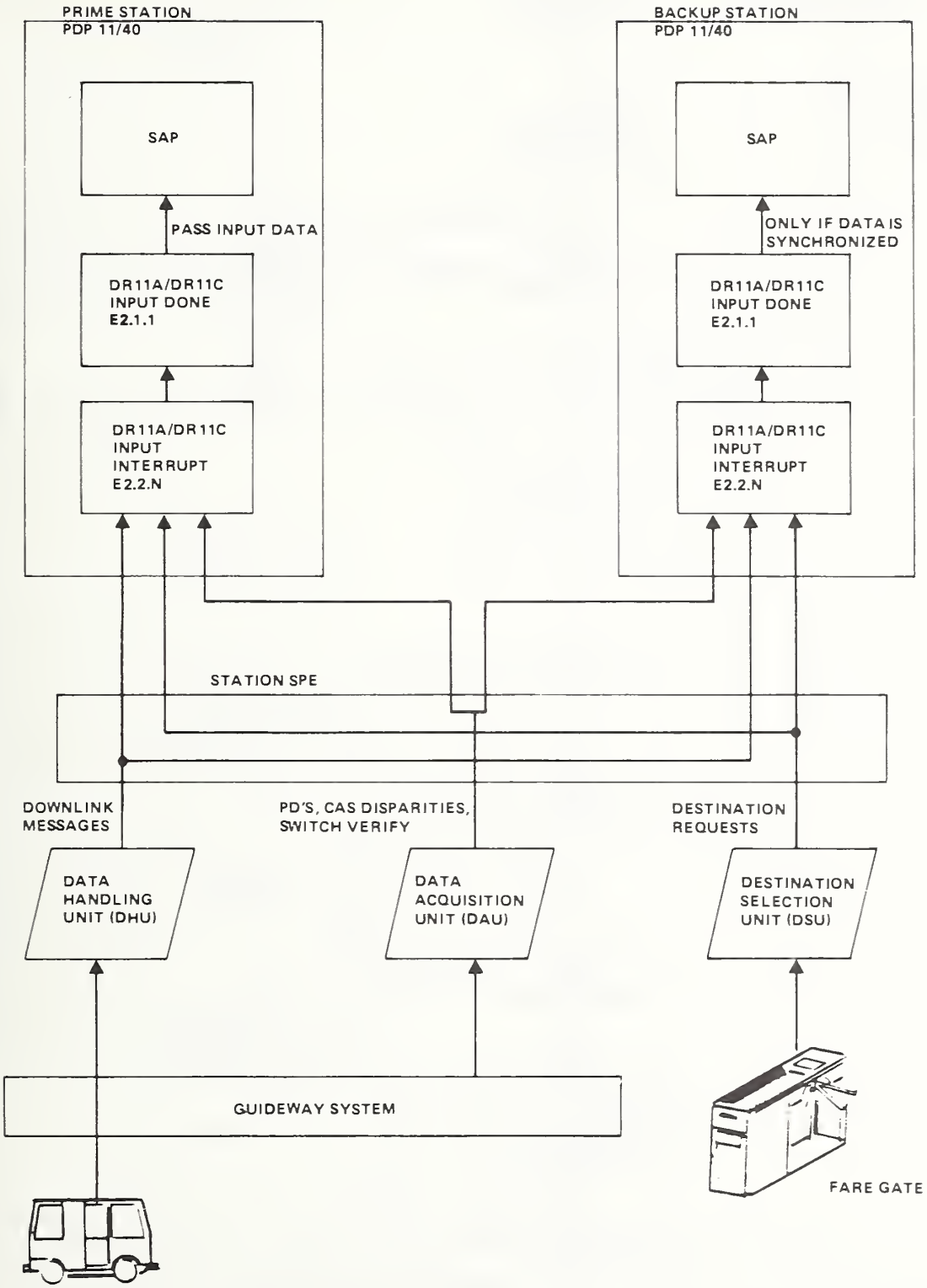


FIGURE 3-19. DR11A/DR11C INPUT DONE CONTROL/DATA FLOW

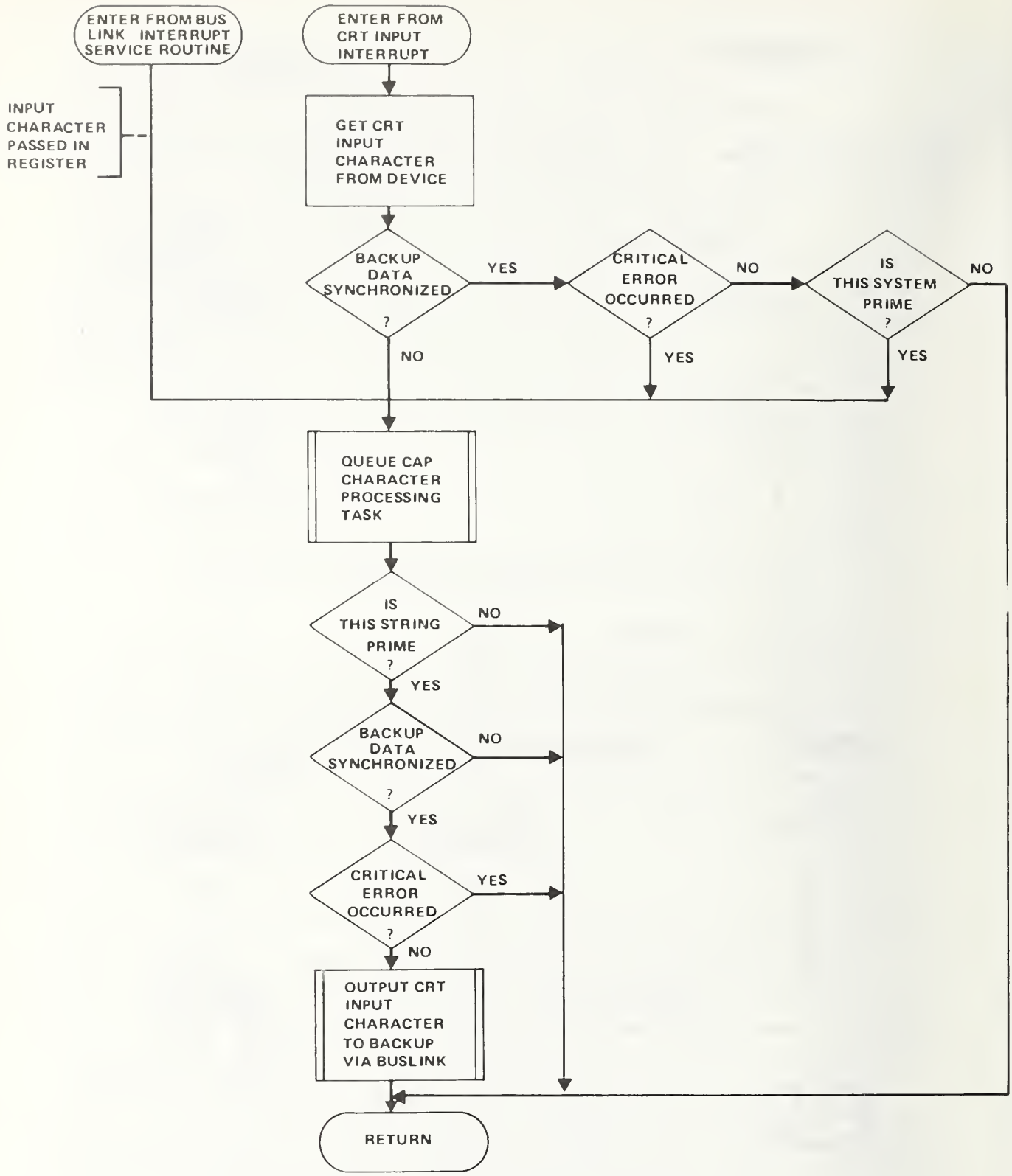


FIGURE 3-20. CRT INPUT INTERRUPT SERVICE

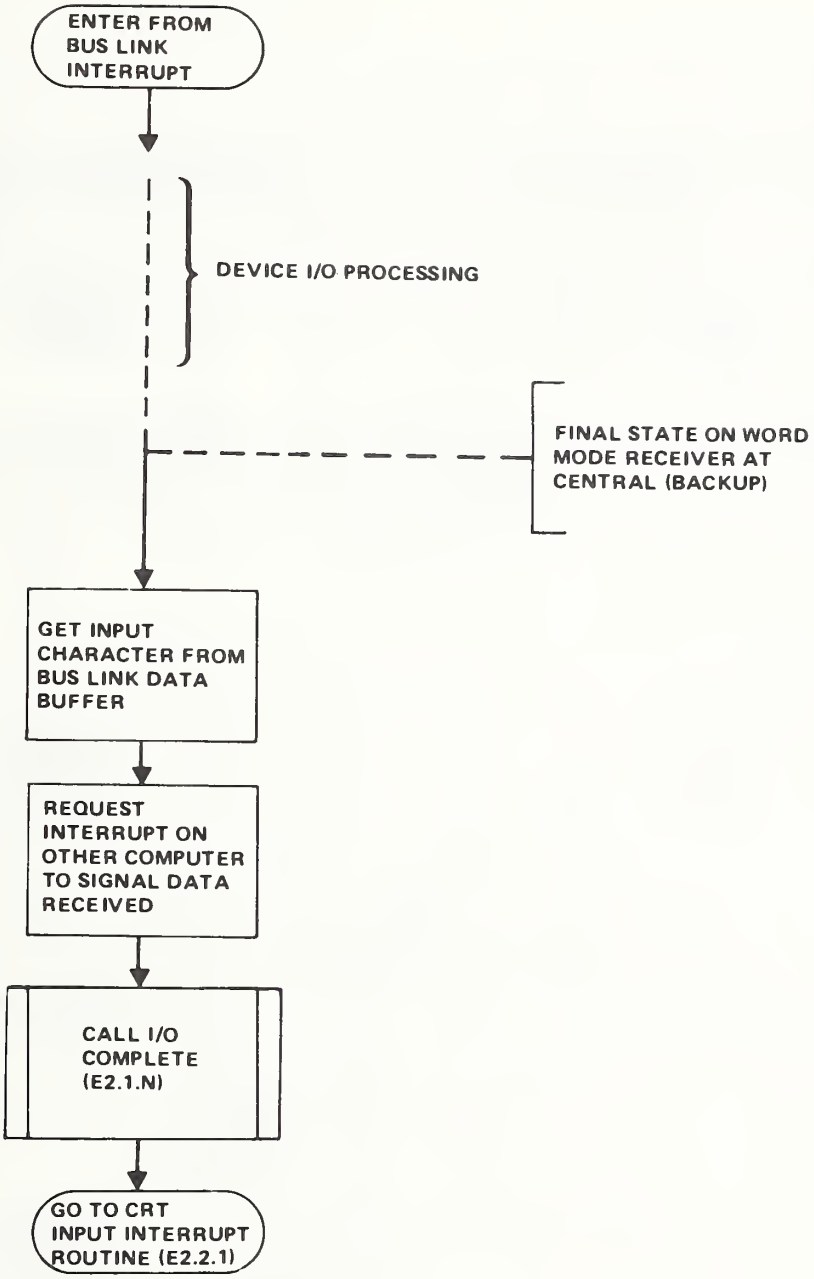

ENTER FROM BUS LINK INTERRUPT
DEVICE I/O PROCESSING
FINAL STATE ON WORD MODE RECEIVER AT CENTRAL (BACKUP)
GET INPUT CHARACTER FROM BUS LINK DATA BUFFER
REQUEST INTERRUPT ON OTHER COMPUTER TO SIGNAL DATA RECEIVED
CALL I/O COMPLETE (E2.1.N)
GO TO CRT INPUT INTERRUPT ROUTINE (E2.2.1)

FIGURE 3-21. BUS LINK INTERRUPT SERVICE

Figure 3-22 shows a control/data flow for the CRT Input Interrupt and Bus Link Interrupt Service routines. This flow shows the condition in which data has been synchronized and the prime CRT input characters are passed to the backup via the bus link. The backup processes and echos the prime CRT characters and ignores the inputs from the backup CRT.

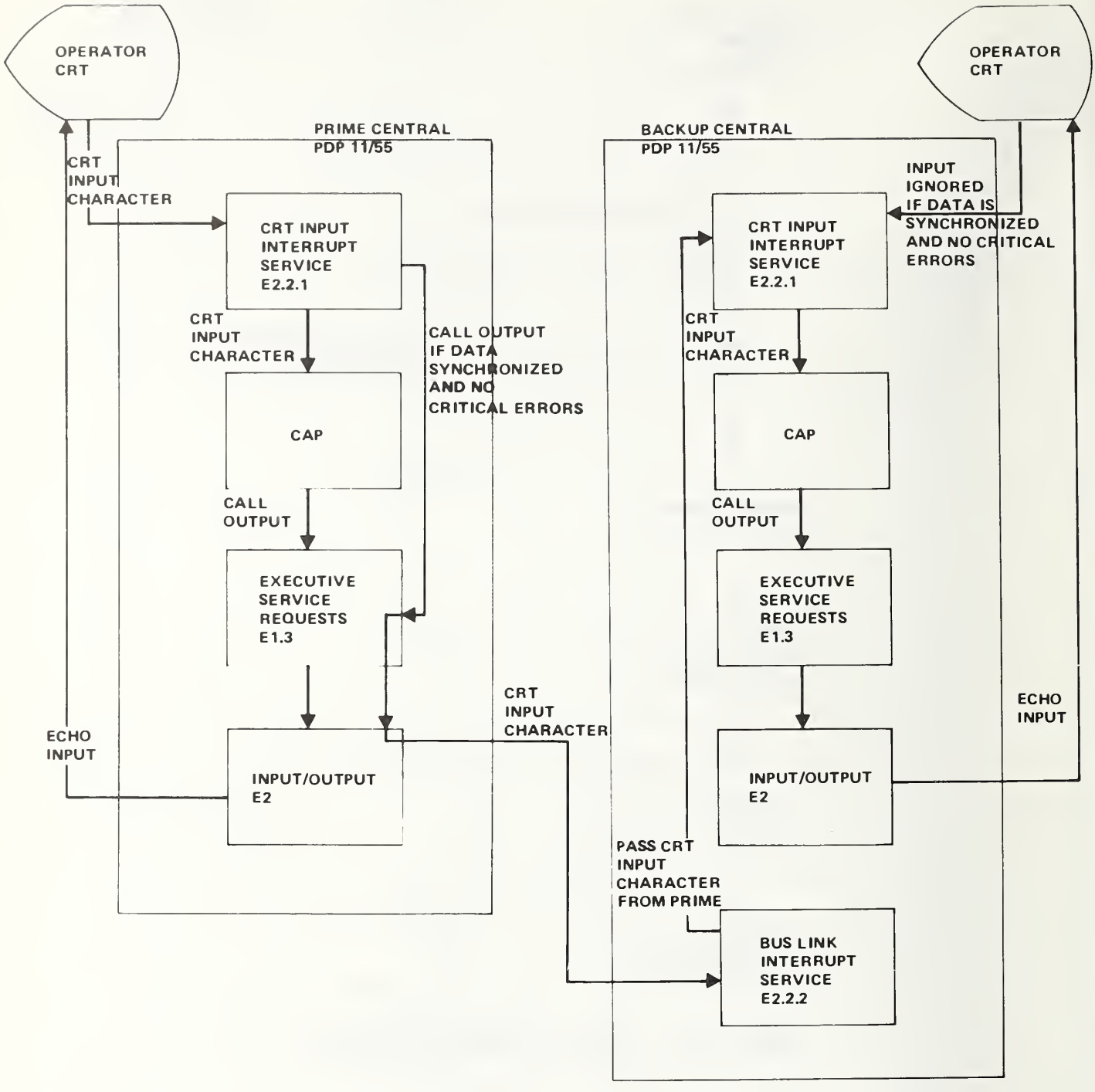


FIGURE 3-22. CRT INPUT INTERRUPT SERVICE AND BUS LNK INTERRUPT SERVICE CONTROL/DATA FLOW

3.3.3 Data Synchronization

This section describes the detailed design of the redundancy functions of the routines in the Data Synchronization module. This module provides for the synchronization of the backup computer string to the prime string. This synchronization educates the backup about the current state of the system and makes the backup string eligible to take control intelligently in case of a prime string failure.

Data Synchronization and Data Synchronization Timeout. The Data Synchronization routine and Data Synchronization Timeout routine perform the data synchronization function. These two routines exist at both central and the stations.

These two routines satisfy the requirement to allow the operator to synchronize the backup system to the state of the prime system with no interference to passenger service. These routines do this by transferring all the executive and application variable data from the prime to the backup string upon operator command. Passenger service is not degraded since no input data is lost and no output commands are suppressed long enough to effect system operation. Since the software and hardware portion of the synchronization process takes less than a second, these two routines satisfy the requirement that the synchronization process must not require the backup to be disarmed for more than 10 seconds.

The process of synchronizing a dual string, real-time control, distributed processing network of fourteen computers with no interruption in control capabilities is quite an accomplishment. It requires coordination throughout the network and between the two strings in a system where the computer clocks do not have sufficient resolution to guarantee the timing required to initiate the data synchronization process. At the completion of synchronization the backup string must be synchronized to the data and processing state of the prime such that the backup operates in parallel with the prime string.

The data synchronization process suspends device I/O and, thus, all communications. One reason for this is that the actual transfer of

the variable data takes approximately 40 ms while a character time on the modems is only 3 ms. Thus, the time for the data transfer would cause modem data overruns and probable communication failures. Another reason is that transferring data which is changing would cause confusion. The synchronization process synchronizes not only the variable data but also the state of the pending interrupts of the machines by temporarily suppressing all I/O. If this were not done, an output in progress on the backup but not on the prime during data synchronization would cause an interrupt on completion that would come as quite a surprise to the backup whose variable data and, thus, I/O buffers had been changed to match the prime. Thus, the suppression of I/O provides a brief quiescent state so that the backup can be totally synchronized to the state of the prime.

One of the primary ground rules during the design of the Data Synchronization module was that failure of the data synchronization process must not degrade the prime string. The Data Synchronization Timeout routine provides a safe-guard against this by timing out the process and aborting it before the prime is adversely affected. Another area of concern for the health of the prime string was bus link failures. Bus link failures can cause the data being overwritten in the backup to be destroyed, thus causing a backup failure. To prevent prime failure, the amount of data and commands written to the prime during data synchronization are kept to a minimum. This approach prevents the prime string software from being destroyed by backup failures or bus link failures.

Figure 3-23 shows the flow of the Data Synchronization routine. As can be seen from the flow, Data Synchronization sends messages throughout the system to start the data synchronization process. Data Synchronization temporarily stops the other central monitoring, the timing out of communications, and device I/O timeouts. This is required since Data Synchronization next signals I/O Management to suppress device I/O.

The Data Synchroniztaion Timeout routine is scheduled to timeout the synchronization process so that if the process either takes too long or fails, the system I/O is restarted before any ill effects occur.

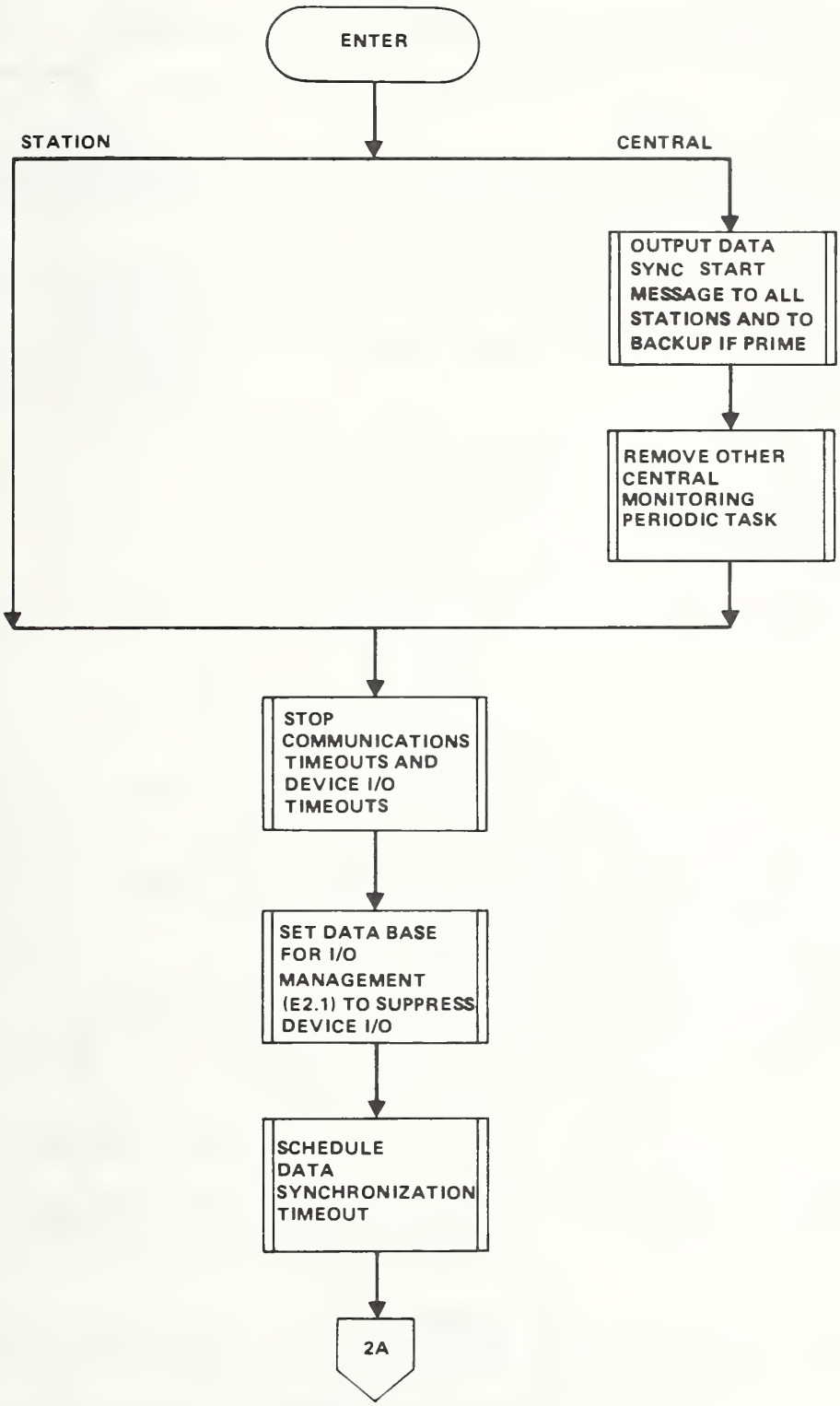

ENTER
STATION
CENTRAL
OUTPUT DATA SYNC START MESSAGE TO ALL STATIONS AND TO BACKUP IF PRIME
REMOVE OTHER CENTRAL MONITORING PERIODIC TASK
STOP COMMUNICATIONS TIMEOUTS AND DEVICE I/O TIMEOUTS
SET DATA BASE FOR I/O MANAGEMENT (E2.1) TO SUPPRESS DEVICE I/O
SCHEDULE DATA SYNCHRONIZATION TIMEOUT
2A

FIGURE 3-23. DATA SYNCHRONIZATION

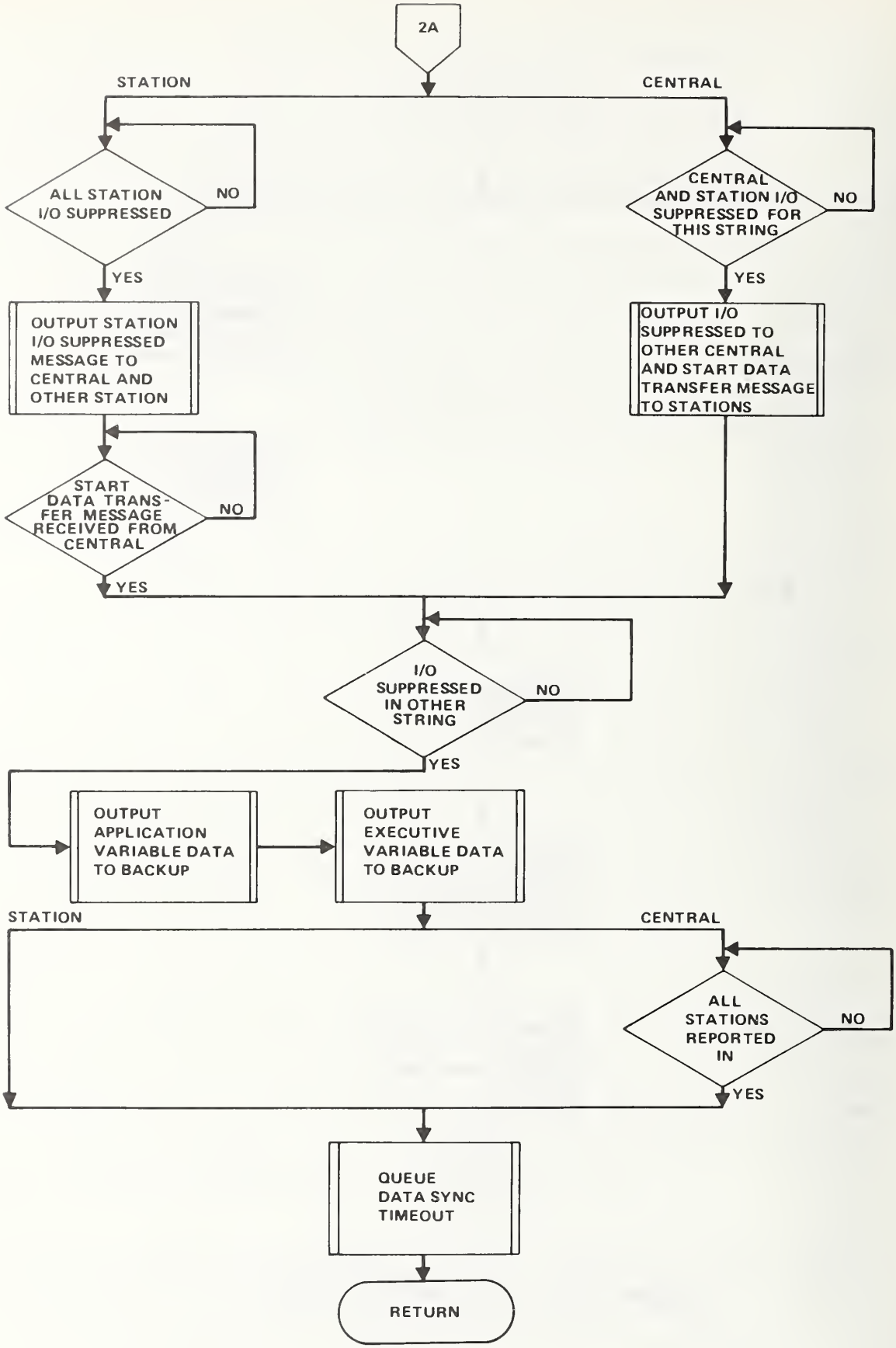


FIGURE 3-23. DATA SYNCHRONIZATION (CONTINUED)

When I/O is suppressed throughout the system, the variable data is transferred from the prime to the backup system, and the Data Synchronization Timeout routine is queued for immediate execution.

Figure 3-24 shows the functional flow of the Data Scynchronization Timeout routine. As can be seen from the flow, this routine cleans up the synchronization process by restarting the I/O devices and timeout functions; in the case of synchronization timeout or failure it also reports the failure to Central/Station Reconfiguration Decision.

Figure 3-25 shows a control/data flow for the Data Synchronization module. This flow gives an indication of the overall system coordination and data transfers required to accomplish the data synchronization process. Notice that Data Synchronization and Daya Synchronization Timeout routines exist in all computers throughout the network.

3.3.4 System Status Monitoring

This section describes the detailed design of the redundancy functions of the routines in the System Status Monitoring module. This module provides for the detection of failures in the computer hardware and software system. Mechanisms used to detect failures include active functional tests, error status tests, reasonableness tests, and process timeouts. Routines in this module also respond to error interrupts issued by the computer hardware. When System Status Monitoring detects a failure, it reports it to the Reconfiguration Control module for the appropriate failure reaction. The System Status Monitoring routines which perform redundancy functions are Other-Central Monitoring and Special Purpose Equipment (SPE) Monitoring.

Other-Central Monitoring. The Other-Central Monitoring routine monitors the viability of the opposite central. This routine exists only at central and consititues the only active other-string monitoring performed by the software.

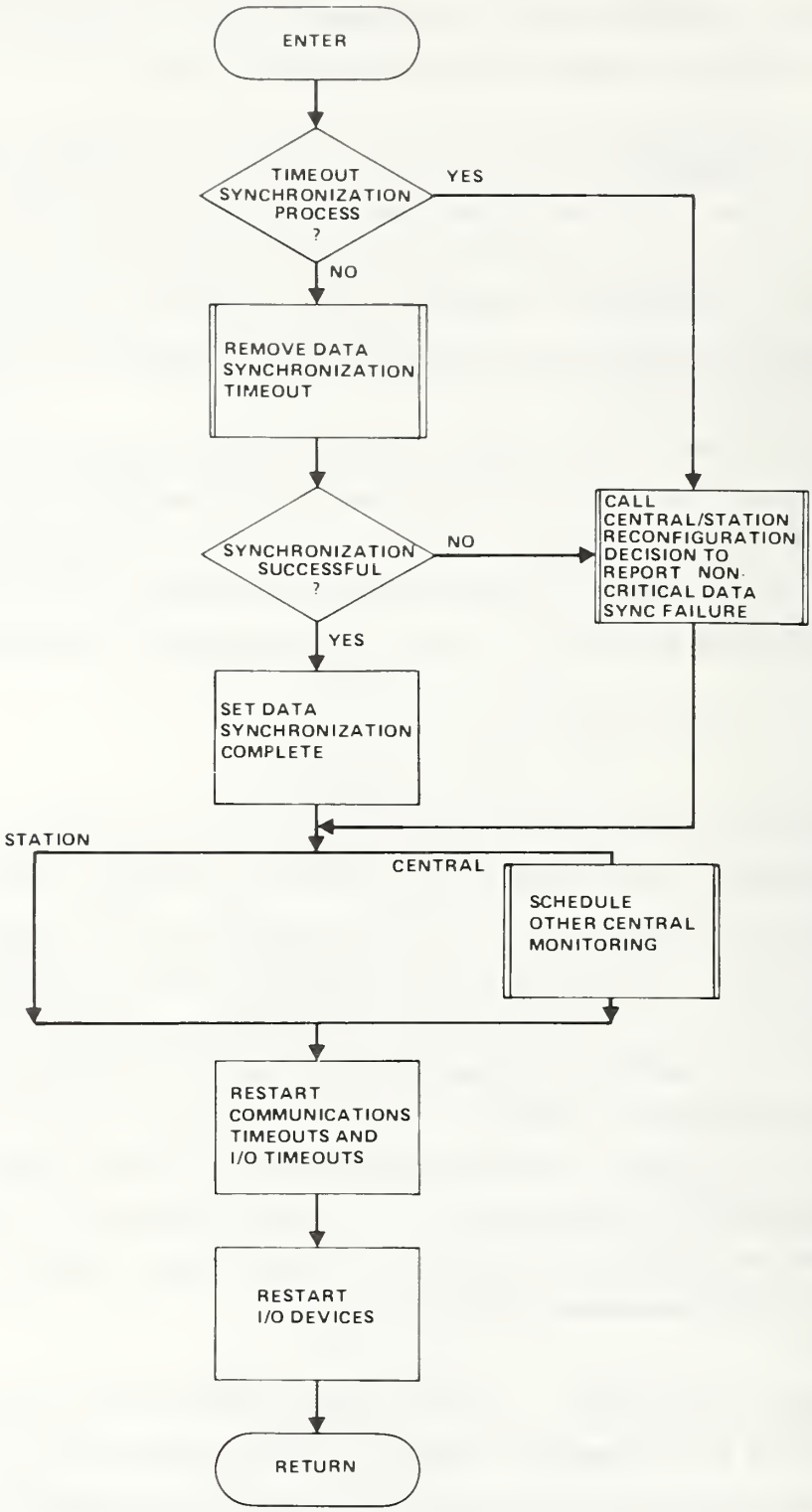


FIGURE 3-24. DATA SYNCHRONIZATION TIMEOUT

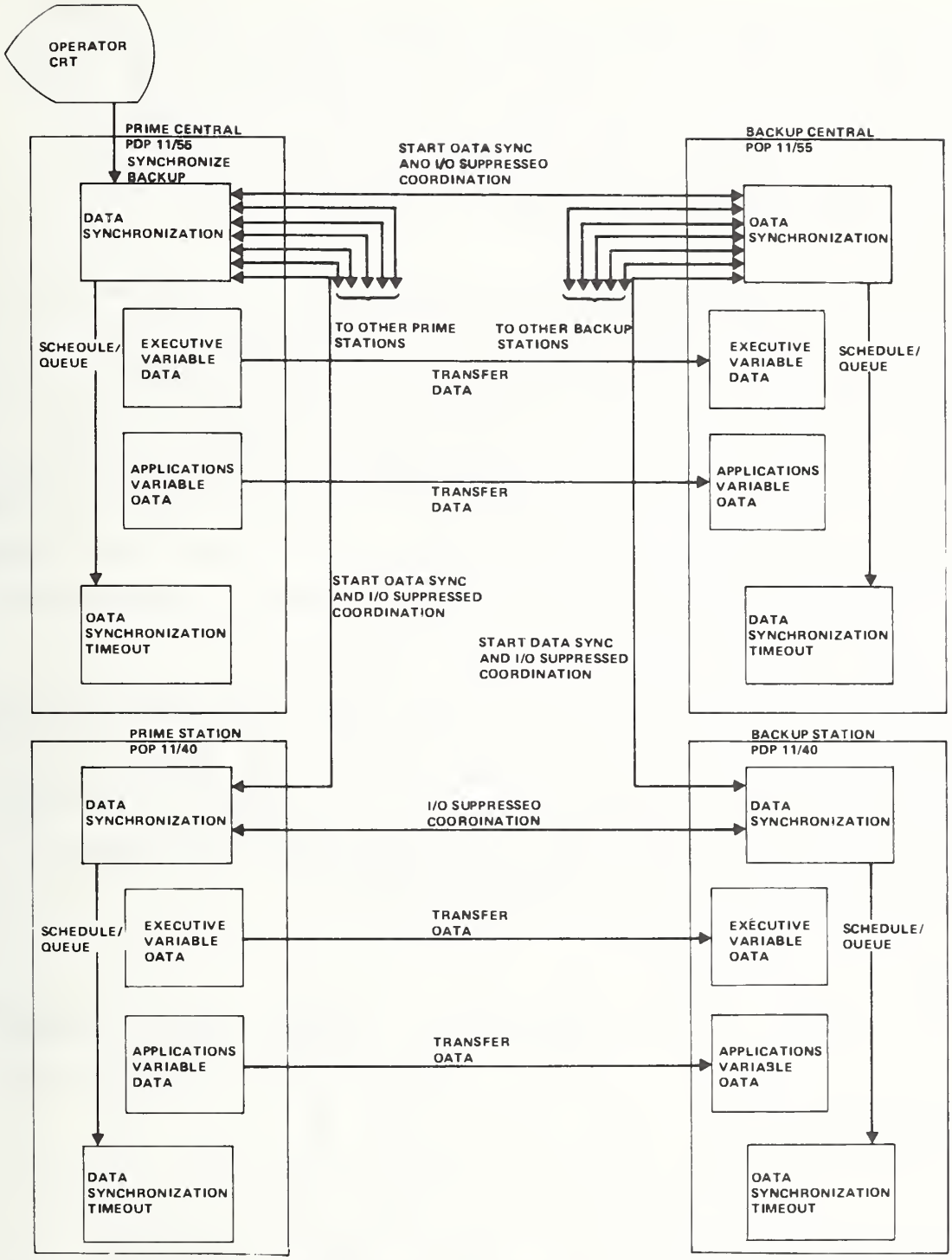


FIGURE 3-25. DATA SYNCHRONIZATION CONTROL/DATA FLOW

The Other-Central Monitoring routine partially satisfies the requirement to detect failures in the redundant computing system in a manner such that reconfigurations can take place with no system down-time. It does this by monitoring the other central for failures by periodically outputting its system status via the bus link to the other central and by timing out the status input from the other central.

Other-Central Monitoring operates once every 100 ms in both the prime and backup central computer. This allows other-central failures to be detected within 300 ms in the worst case. Three times the period of the routine is required because of the case illustrated in Figure 3-26. In this case the executions are phased very closely, and the executions in the prime are delayed occasionally by interrupt processing. For this reason Other-Central Monitoring can operate twice on one string with no intervening Other-Central Monitoring execution on the opposite string. Thus, Other-Central Monitoring must give the other string two chances before declaring a timeout. Again referring to the case in Figure 3-26, the prime could fail right after Other-Central Monitoring execution, and the other central timeout would not be detected until the third execution in the backup as shown. Even though this detection mechanism may take up to 300 ms, it still provides failure detection and reconfiguration within the required 500 ms.

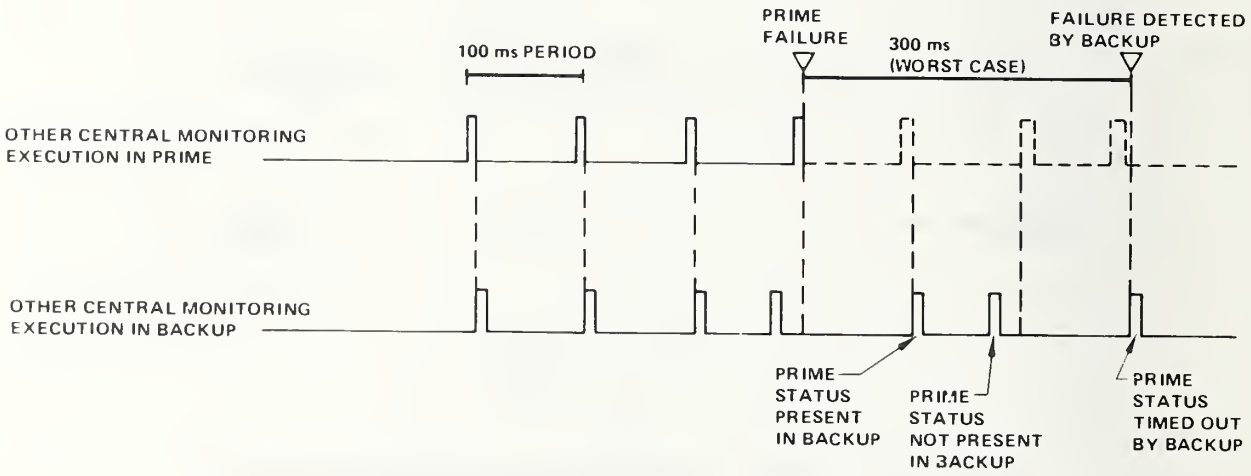

100 ms PERIOD
PRIME FAILURE
300 ms (WORST CASE)
FAILURE DETECTED BY BACKUP
OTHER CENTRAL MONITORING EXECUTION IN PRIME
OTHER CENTRAL MONITORING EXECUTION IN BACKUP
PRIME STATUS PRESENT IN BACKUP
PRIME STATUS NOT PRESENT IN BACKUP
PRIME STATUS TIMED OUT BY BACKUP

FIGURE 3-26. WORST CASE OTHER-CENTRAL FAILURE DETECTION TIME LINE

Figure 3-27 shows the functional flow of the Other-Central Monitoring routine. As can be seen from the flow, once a critical error has occurred other-central monitoring is stopped. This prevents other-central timeouts from being reported repeatedly. The timeout function is started smoothly by having the backup send an initial message to the prime which, in turn, echos it. When each side receives the initial message it sets its string to dual to start the other-central timeout function. Once in dual mode, Other-Central Monitoring outputs its own string's system status and times out the input of the other string's status. When the input from a string is timed out, Other-Central Monitoring uses the secondary communications link to decide which string is going to seize control. If the secondary link was not used for arbitration on other-central timeouts, a single point failure in the bus link could cause system failure. A bus link failure when both strings are still operable would cause each string to time out the other and attempt to seize control. This would result in both strings thinking they are the prime controlling string and could result in some of the station SPEs to be set to one string and some to be set to the other making the system inoperable. However, the use of the secondary communications link between the two centrals allows Other-Central Monitoring to resolve simultaneous other-central timeouts with the backup yielding control in case of a tie. The tie occurs when both the prime and the backup are executing page two of Figure 3-27 at the same time. The string which wins the race reports an "other-central timeout" to Central Reconfiguration Decision. The string which loses reports a "shutdown this computer string" failure message.

Figure 3-28 shows a control/data flow for Other-Central Monitoring. As can be seen from this figure, this routine uses the central-to-central bus link via the Output ESR to transfer status between the central computers. If this communication breaks down, Other-Central Monitoring uses the secondary communication link to arbitrate the other-central timeout. Failures are reported to Central Reconfiguration Decision which acts on the failures to effect the proper reconfiguration.

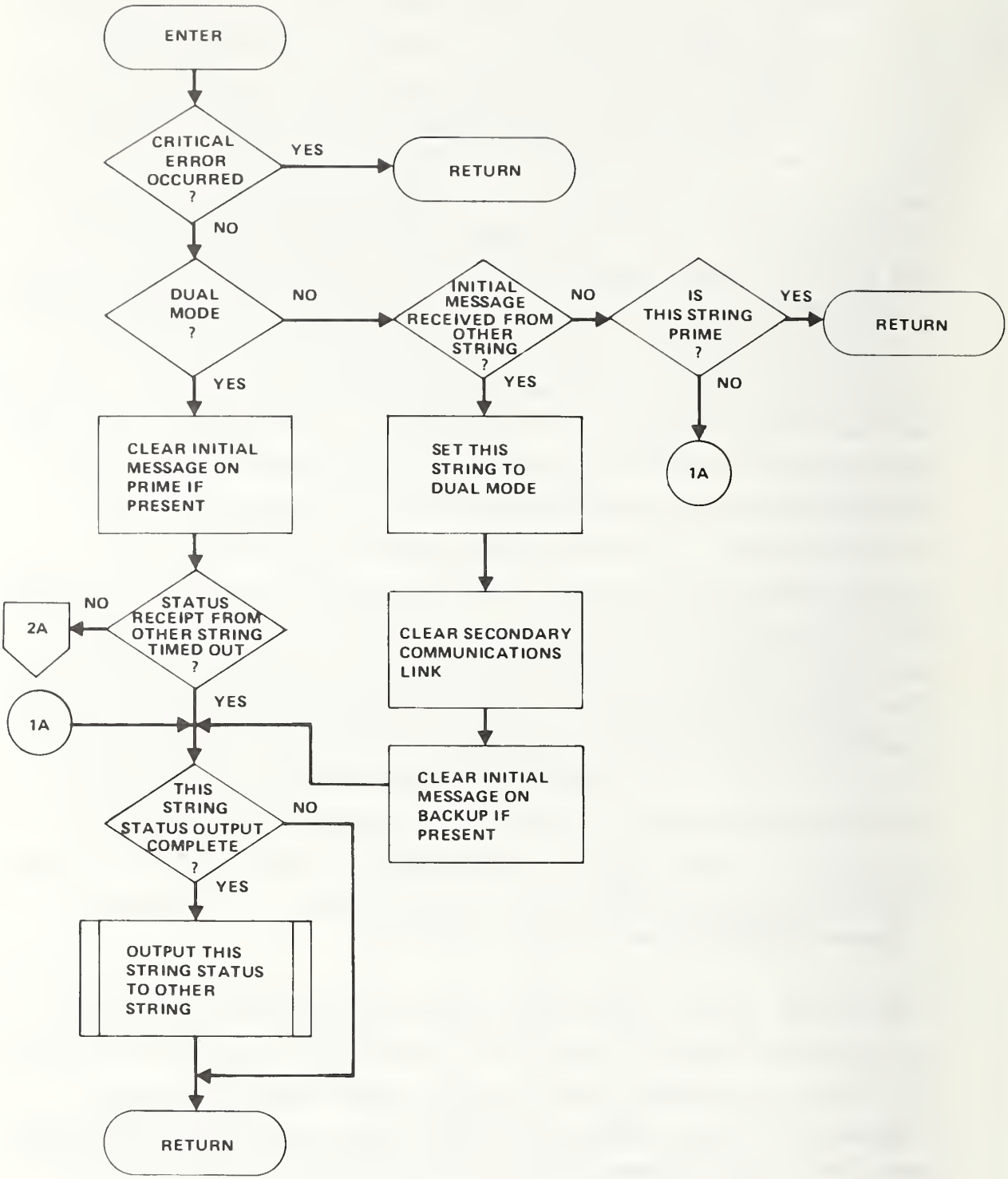


FIGURE 3-27. OTHER-CENTRAL MONITORING

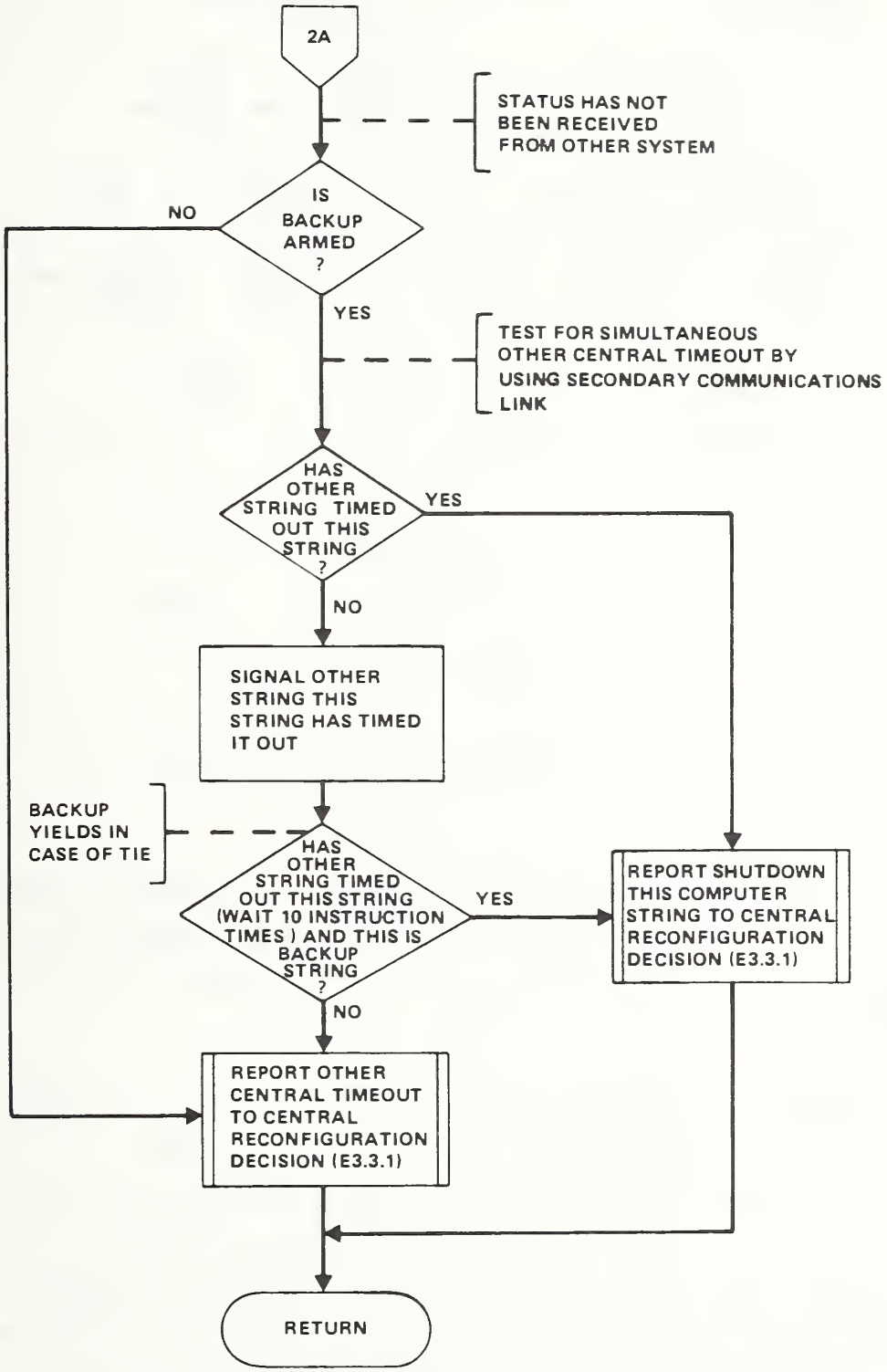

2A
STATUS HAS NOT BEEN RECEIVED FROM OTHER SYSTEM
IS BACKUP ARMED ?
NO
YES
TEST FOR SIMULTANEOUS OTHER CENTRAL TIMEOUT BY USING SECONDARY COMMUNICATIONS LINK
HAS OTHER STRING TIMED OUT THIS STRING ?
YES
NO
SIGNAL OTHER STRING THIS STRING HAS TIMED IT OUT
BACKUP YIELDS IN CASE OF TIE
HAS OTHER STRING TIMED OUT THIS STRING (WAIT 10 INSTRUCTION TIMES) AND THIS IS BACKUP STRING ?
YES
NO
REPORT SHUTDOWN THIS COMPUTER STRING TO CENTRAL RECONFIGURATION DECISION (E3.3.1)
REPORT OTHER CENTRAL TIMEOUT TO CENTRAL RECONFIGURATION DECISION (E3.3.1)
RETURN

FIGURE 3-27. OTHER-CENTRAL MONITORING (CONTINUED)

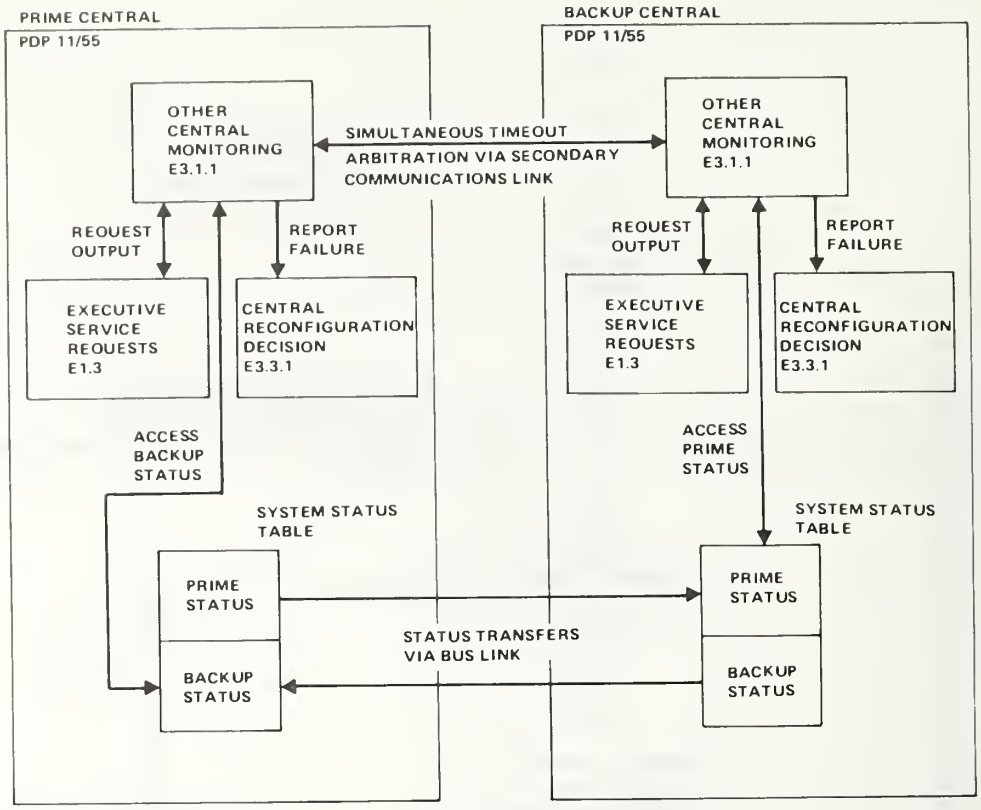


FIGURE 3-28. OTHER-CENTRAL MONITORING CONTROL/DATA FLOW

Special Purpose Equipment (SPE) Monitoring. The Special Purpose Equipment Monitoring routine responds to DHU, DAU, and DSU station SPE timeout interrupts. The station SPE and, thus, this routine exists at station only.

The SPE Monitoring routine partially satisfies the requirement to monitor the system for failures and set SPE to the proper mode. If a station computer fails to respond to a DHU, DAU, or DSU input interrupt within 50 ms after the other computer has responded, the SPE fires a SPE timeout interrupt to the "good" computer, the computer which answered the input interrupt. The SPE Monitoring routine responds to these SPE timeouts and in certain cases seizes SPE in prime single mode to the good computer.

Figure 3-29 shows the functional flow of the SPE Monitoring routine. As can be seen from this flow, when this routine is entered from a SPE timeout, it queues Station Reconfiguration Decision to report the

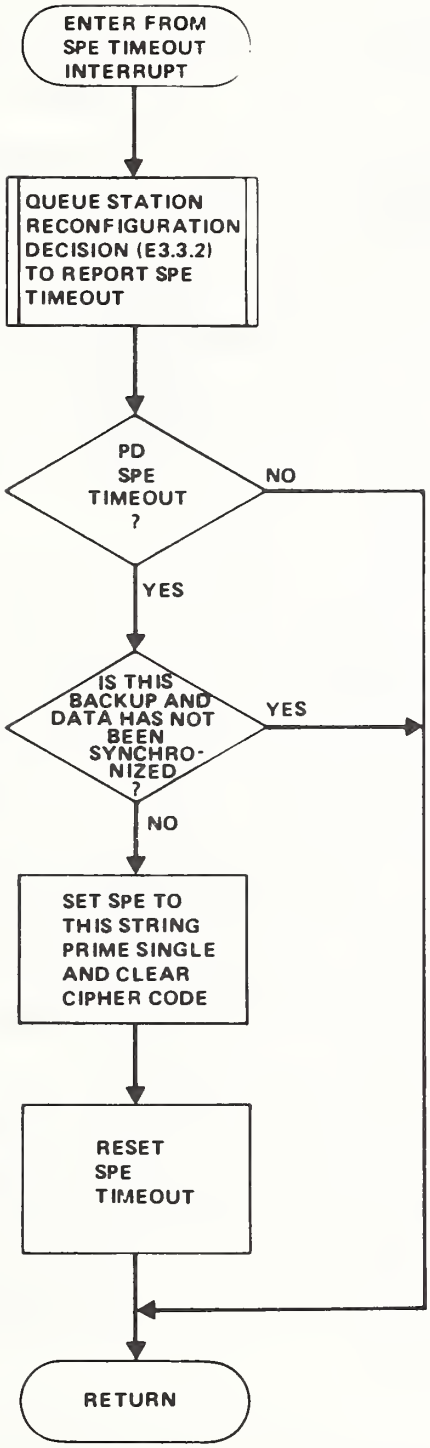

ENTER FROM SPE TIMEOUT INTERRUPT
QUEUE STATION RECONFIGURATION DECISION (E3.3.2) TO REPORT SPE TIMEOUT
PD SPE TIMEOUT ?
NO
YES
IS THIS BACKUP AND DATA HAS NOT BEEN SYNCHRO-NIZED ?
YES
NO
SET SPE TO THIS STRING PRIME SINGLE AND CLEAR CIPHER CODE
RESET SPE TIMEOUT
RETURN

FIGURE 3-29. SPECIAL PURPOSE EQUIPMENT (SPE) MONITORING

SPE timeout. SPE monitoring then decides if it should seize the SPE. The approach here is to have central make the reconfiguration decision in all cases possible. Normally, a SPE timeout signals that the other computer has failed and cannot operate the system meaning that the SPE Monitoring routine could always seize the SPE. However, SPE timeouts do occur when the other computer could still operate the system. If the SPE Monitoring routine always seized SPE, it could seize the SPE away from a good prime to an unarmed backup causing systems stoppage. For this reason all SPE timeouts except PD SPE timeouts are reported to central so that central can decide who should seize control of the system.

PD SPE timeouts are handled at the station because timing requirements for this device do not always allow time for the failure to be communicated to central and the reconfiguration command to be communicated back to the station before a 500 ms CAS disparity occurs. Thus, SPE Monitoring seizes the SPE on PD SPE timeouts except in the backup system when data has not been synchronized. In this case the backup cannot possibly be armed since it is not synchronized so that the backup should not seize the SPE. The decision to seize the SPE on PD SPE timeouts on the backup should ideally be made based on backup armed status. However, backup armed status does not exist at the stations. Data synchronization status is used as a clue to the armed status since most of the time when the backup is data synchronized, it is also armed. The decision not to maintain backup armed status at the stations is arbitrary. It was decided not to add this capability since the case in which it is needed does not occur very often. This case occurs when a PD SPE timeout is given to a synchronized but not armed backup while the prime station is still operable.

Figure 3-30 shows a control/data flow for the SPE Monitoring routine. This flow depicts the case in which the prime has failed and the SPE has fired a SPE timeout for one of the three input devices to the backup station. The SPE Monitoring routine reports the failure to Station

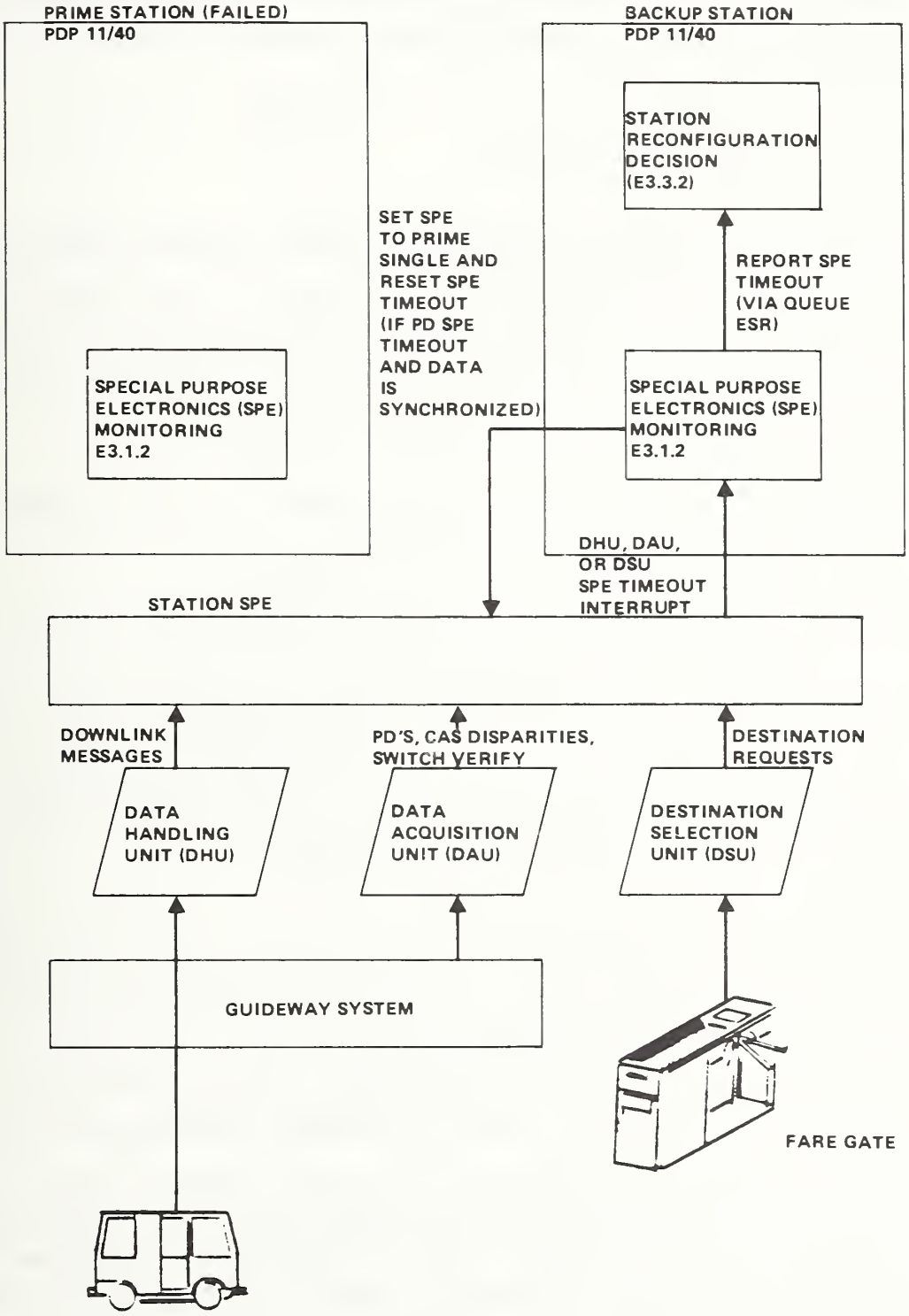

PRIME STATION (FAILED)
PDP 11/40
BACKUP STATION
PDP 11/40
STATION RECONFIGURATION DECISION (E3.3.2)
SET SPE TO PRIME SINGLE AND RESET SPE TIMEOUT (IF PD SPE TIMEOUT AND DATA IS SYNCHRONIZED)
REPORT SPE TIMEOUT (VIA QUEUE ESR)
SPECIAL PURPOSE ELECTRONICS (SPE) MONITORING E3.1.2
SPECIAL PURPOSE ELECTRONICS (SPE) MONITORING E3.1.2
DHU, DAU, OR DSU SPE TIMEOUT INTERRUPT
STATION SPE
DOWNLINK MESSAGES
PD'S, CAS DISPARITIES, SWITCH VERIFY
DESTINATION REQUESTS
DATA HANDLING UNIT (DHU)
DATA ACQUISITION UNIT (DAU)
DESTINATION SELECTION UNIT (DSU)
GUIDEWAY SYSTEM
FARE GATE

FIGURE 3-30. SPECIAL PURPOSE EQUIPMENT (SPE) MONITORING CONTROL/DATA FLOW

Reconfiguration Decision and, in the cases discussed, seizes the station SPE to prime single mode. Normally, Station Reconfiguration Decision would report the failure to central who would command a seize control to the entire backup system.

3.3.5 Configuration Control

This section describes the detailed design of the redundancy functions of the routines in the Configuration Control module. The Configuration Control module performs the system configuration, loading, and initialization functions. The only Configuration Control routine which is affected by dual string is the Central Loader.

Central Loader. The Central Loader routine loads the central computer and directs the overall string loading process.

The Central Loader routine satisfies the requirement to allow either string to be loaded as prime or backup and partially satisfies the requirement to maintain the prime/backup status for display to the operator and reconfiguration decisions. It does this by inputting prime/backup status from the operator and setting the executive data base to prime or backup as directed by the operator.

Figure 3-31 shows the functional flow and Figure 3-32 shows a control/data flow of the Central Loader routine. As can be seen from these flows, the Central Loader directs the loading of the computer string then inputs the prime/backup status from the operator and sets the status into the central executive data base. This status is transmitted to the stations in each modem message. When a status change is received at the stations the modem communication routines call Station Reconfiguration Processing to process that change. This is the mechanism through which prime/backup status and also dual/single status is distributed throughout a computer string.

As explained in a previous section, the same software image is loaded into both the A and B string, and the software has no knowledge in

which physical string it resides. The prime/backup status set by the Central Loader enables the software to determine whether it is the prime or backup as defined by initial operator input.

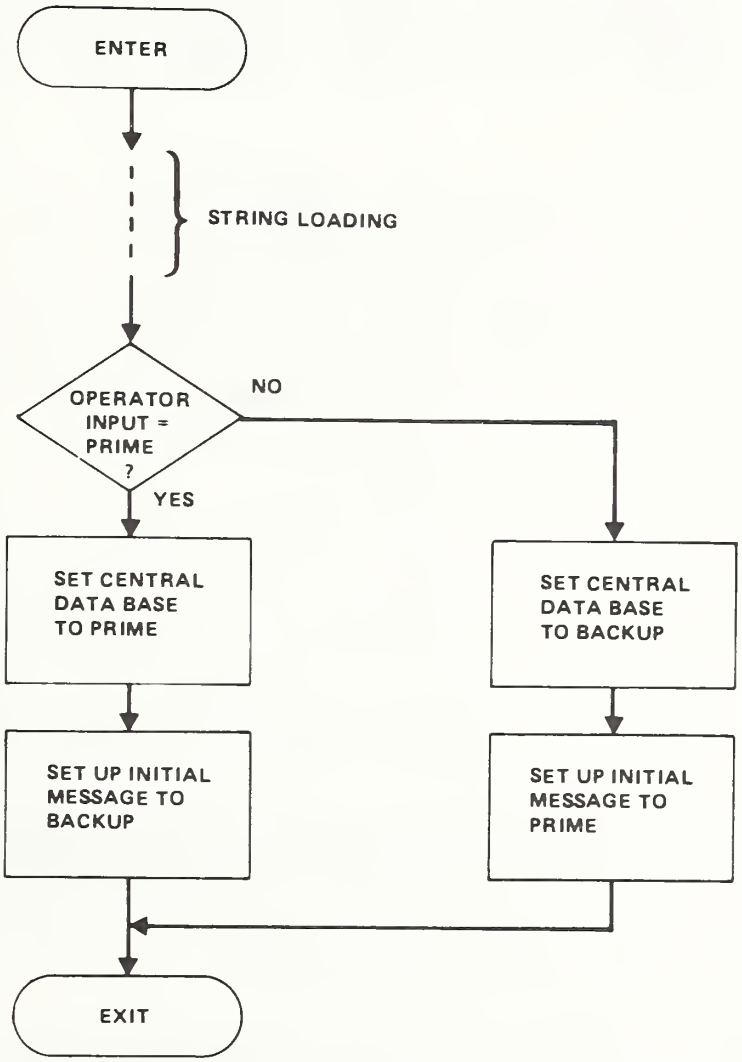


FIGURE 3-31. CENTRAL LOADER

3.3.6 Reconfiguration Control

This section describes the detailed design of the redundancy functions of the routines in the Reconfiguration Control module. This module makes reconfiguration decisions and performs the reconfigurations required by changes in the system environment such as loading of the backup, or failures in the prime or backup strings. Reconfigurations include actions such as the change from single to dual mode, switchover to the backup, seize control by the prime in prime single mode, or system shutdown.

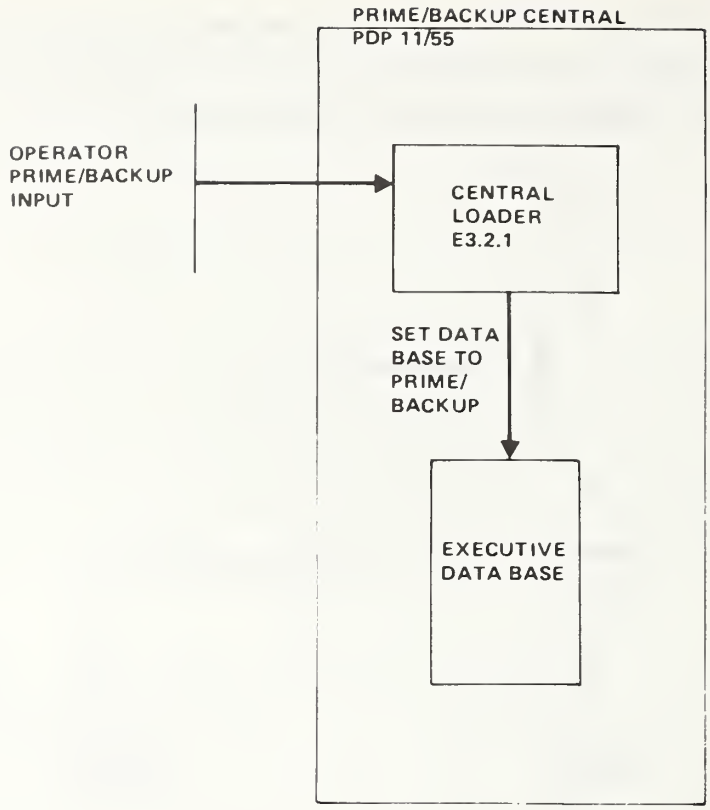


FIGURE 3-32. CENTRAL LOADER CONTROL/DATA FLOW

Central Reconfiguration Decision and Central Reconfiguration Processing. The Central Reconfiguration Decision and Processing routines decide the proper reconfiguration required to react to failures and perform the central portion of the reconfiguration.

The Central Reconfiguration Decision and Processing routines satisfy the requirement to react to failures by performing reconfigurations automatically. They do this by considering each failure and the current state of the system to determine the proper reconfiguration after which they perform the central portion of the reconfiguration and command their counterpart in the stations to do the station portion of the reconfiguration. Central Reconfiguration Processing satisfies the requirement to inform the operator of reconfigurations and changes in prime/backup status by queueing the application switchover task on reconfigurations to display the required notification. Central Reconfiguration Decision satisfies the important safety requirement to remove guideway power in the event of failures in the redundant

computing system which could endanger pasenger safety. It does this by tripping power for critical failures when it cannot successfully perform a reconfiguration yielding an unfailed computing string. It also reports pseudo-non-critical failures to the application software which trips power in the control zone of the station where the failure occurred.

All failures in the computing system are reported to Central Reconfiguration Decision. As shown in Figure 3-33, this routine determines the proper reconfiguration based on the type of failure and the current state of the system. Central Reconfiguration Decision in the prime string reacts to critical failures by reporting the failure to the backup if it is armed or by tripping power if the backup is not armed. The prime reacts to critical other-string failures by calling Central Reconfiguration Processing to seize control in prime single mode. Central Reconfiguration Decision in the backup reacts to critical this-string failures by reporting them to the prime. The backup reacts to critical other-string failures if it is armed by calling Central Reconfiguration Processing to seize control or by only reporting the failure to the central operator if it is not armed. Central Reconfiguration Decision records in the executive data base (posts) the first critical this-string failure or if no critical errors have occurred, the last non-critical error which occurred. Thus, any executive routine can determine if a critical this-string failure has occurred on the current load. Central Reconfiguration Decision uses this capability to detect dual failures. When processing other-string critical failures, Central Reconfiguration Decision checks to see if a this-string critical error has occurred. If one has, then a critical failure has occurred in both the prime and the backup systems, and a dual failure has taken place. If a critical this-string failure has not occurred, then it is all right to seize control.

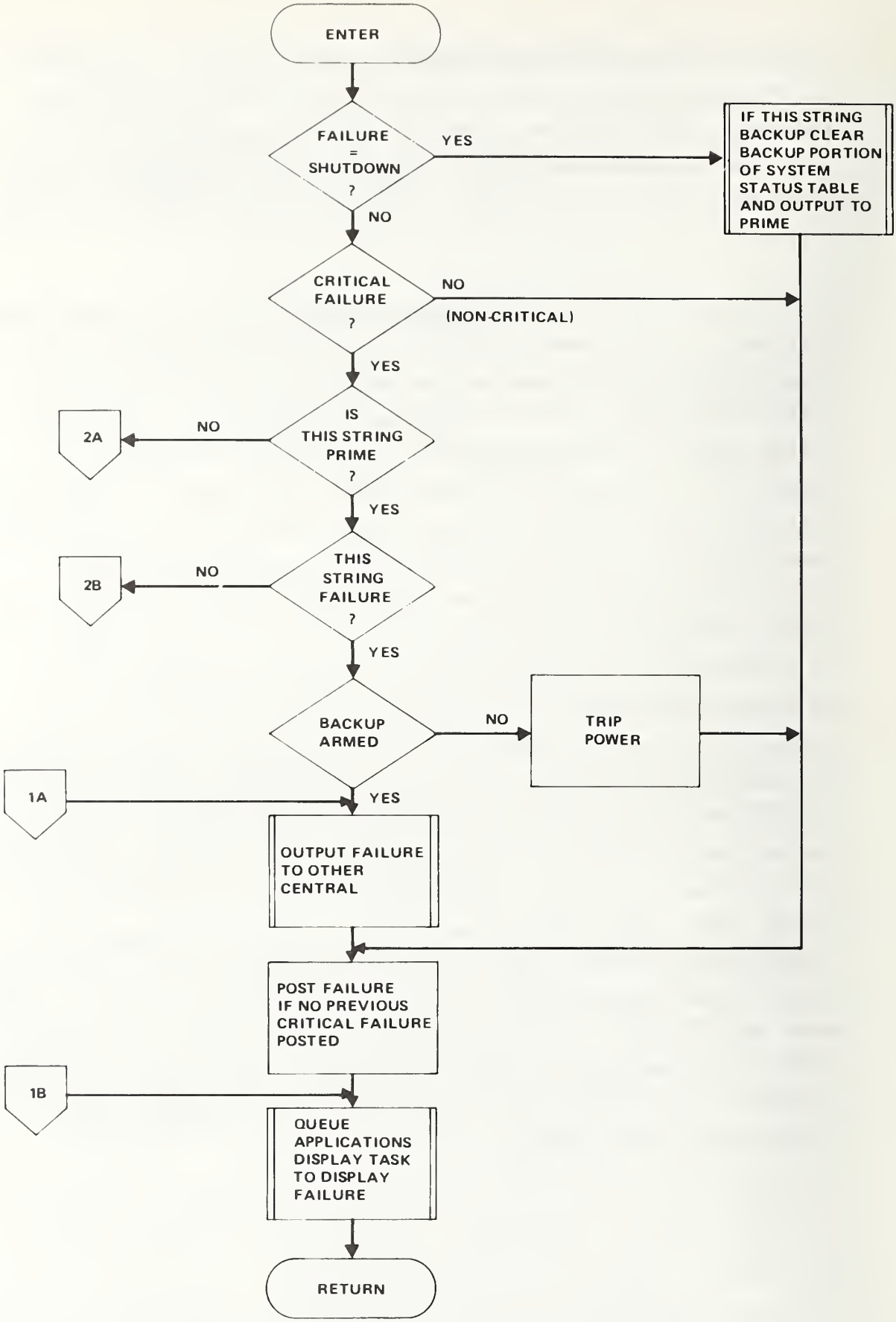


FIGURE 3-33. CENTRAL RECONFIGURATION DECISION

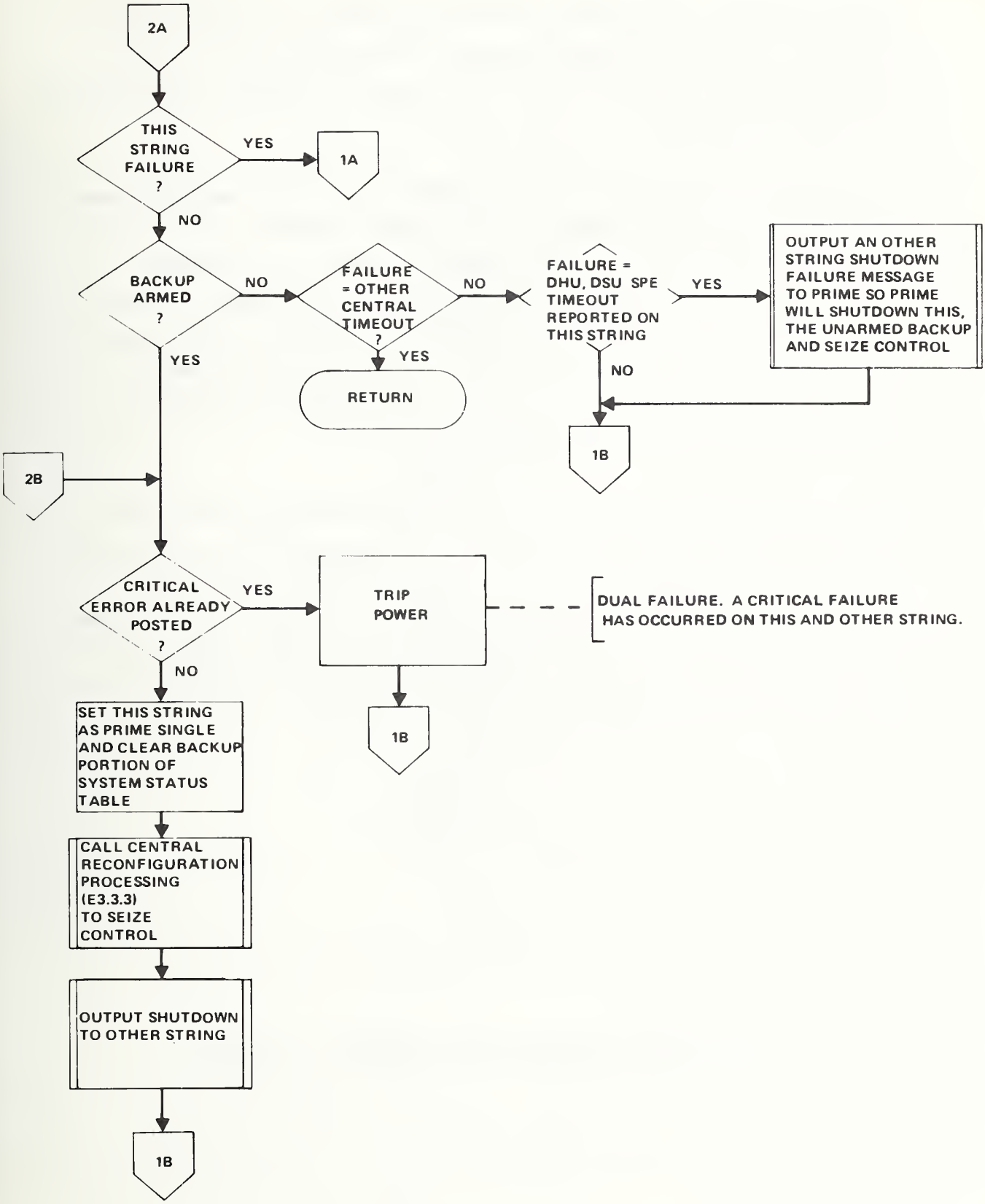


FIGURE 3-33. CENTRAL RECONFIGURATION DECISION (CONTINUED)

Whenever Central Reconfiguration Decision determines it must seize control, it outputs a shutdown message to the other central. This informs the other central that it is no longer an unfailed string eligible to control the system.

The categorizing of failures into critical/non-critical, and this/other-string works very well except for SPE timeouts reported to an unarmed backup indicating failure of the prime to respond to an interrupt. In this case, it is better for the unarmed backup to report an other-string shutdown and force the prime to assert itself as prime single switching the SPEs to the prime string. This reaction performed by Central Reconfiguration Decision has the best chance of continuing system operation without interruption.

When Central Reconfiguration Decision determines that it must seize control, it calls Central Reconfiguration Processing to carry out the mechanics of the actual seizure. As shown in Figure 3-34, Central Reconfiguration Processing sets the executive data base status to prime single, outputs a seize control message to each station, queues the applications switchover task to inform the operator of the seizure, and calls Activate SPE ESR to seize the central SPE.

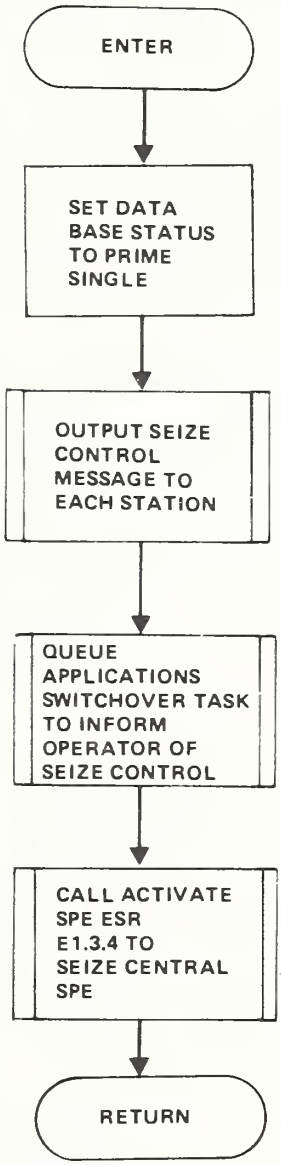

ENTER
SET DATA BASE STATUS TO PRIME SINGLE
OUTPUT SEIZE CONTROL MESSAGE TO EACH STATION
QUEUE APPLICATIONS SWITCHOVER TASK TO INFORM OPERATOR OF SEIZE CONTROL
CALL ACTIVATE SPE ESR E1.3.4 TO SEIZE CENTRAL SPE
RETURN

FIGURE 3-34. CENTRAL RECONFIGURATION PROCESSING

Figure 3-35 shows a control/data flow for Central Reconfiguration Decision and Central Reconfiguration Processing. This flow depicts the case in which a failure in the prime string is reported to the prime Central Reconfiguration Decision which, in turn, directs Central Reconfiguration Processing to seize control in prime single mode.

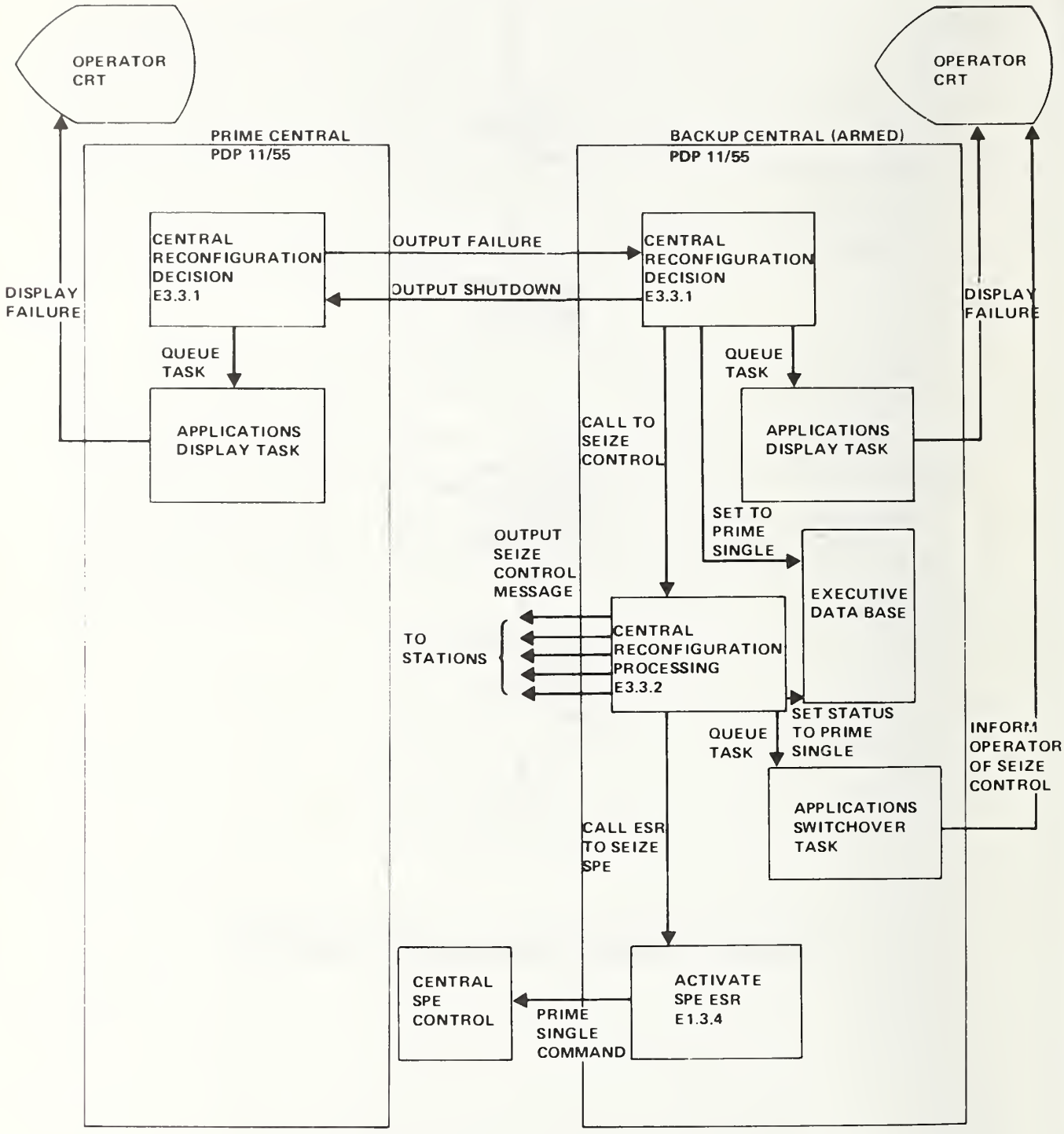


FIGURE 3-35. CENTRAL RECONFIGURATION DECISION AND PROCESSING CONTROL/DATA FLOW

Station Reconfiguration Decision and Station Reconfiguration Processing. The Station Reconfiguration Decision and Processing routines report failures to central and respond to the changes in the system configuration as dictated by central.

Station Reconfiguration Decision assists in satisfying the requirement to reconfigure automatically by reporting failures to Central Reconfiguration Decision. Station Reconfiguration Processing satisfies the requirement to set the stations SPEs to the proper mode by controlling the station SPEs as directed by the modem communication status changes from central.

Figure 3-36 shows the functional flow of the Station Reconfiguration Decision routine. As can be seen from this flow, Station Reconfiguration Decision reports failures to central and records this-string failures.

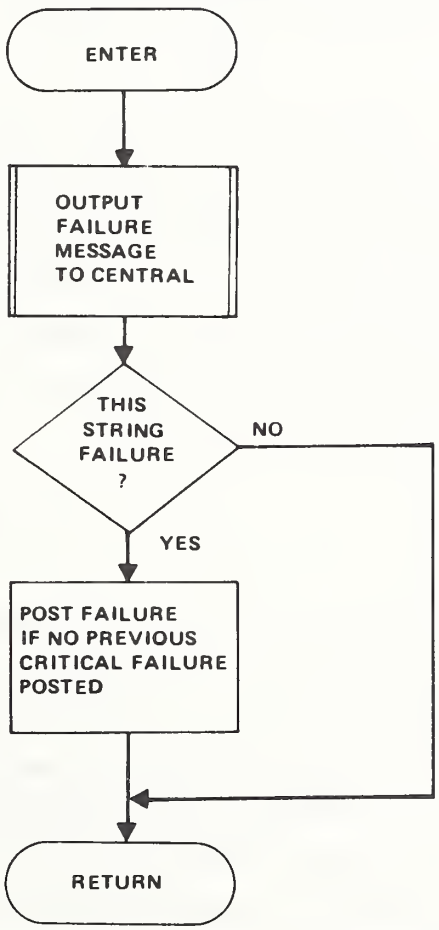


FIGURE 3-36. STATION RECONFIGURATION DECISION

Figure 3-37 shows the functional flow of Station Reconfiguration Processing. As can be seen from this flow, Station Reconfiguration Processing controls the station SPEs and sets the station executive data base prime/backup, single/dual status based on the status in the modem communication messages. The setting of station SPE on the initial loading of the prime is not performed by Station Reconfiguration Processing but is deferred and performed by the Activate SPE ESR when called on the activate I/O central operator command. This provides for a more orderly startup sequence and prevents CAS disparities in many cases. Except for this initial loading case and for the the case of PD SPE timeouts explained previously, all station SPE control is performed by Station Reconfiguration Processing. The flow in Figure 3-37 also shows that Station Reconfiguration Processing queues an applications switchover task on seize controls. This applications task outputs the last DHU uplink commands issued before the SPE change thereby reinforcing these commands. This is required since the backup string output commands are inhibited by SPE and because the prime may have failed to output the last commands.

Figure 3-38 shows a control/data flow for Station Reconfiguraton Decision and Processing. This flow depicts the case where a failure in a prime station is reported to Station Reconfiguration Decision. This routine reports the failure to central. This results in a status change to prime single mode and a seize control message to the backup station. The backup Station Reconfiguration Processing seizes control in prime single mode based on this message input.

3.4 Off-The-Shelf Versus Special Design Components

At the time the MPM system was built, there were no off-the-shelf redundant computing systems available which satisfied even a minimum subset of the system requirements. However, the MPM redundant computing system was built using as many off-the-shelf components as feasible. This section identifies the redundant system components which were purchased off-the-shelf and those which were specially built for the MPM system. This section also discusses the availability of off-the-shelf components for the building of a redundant computing system today.

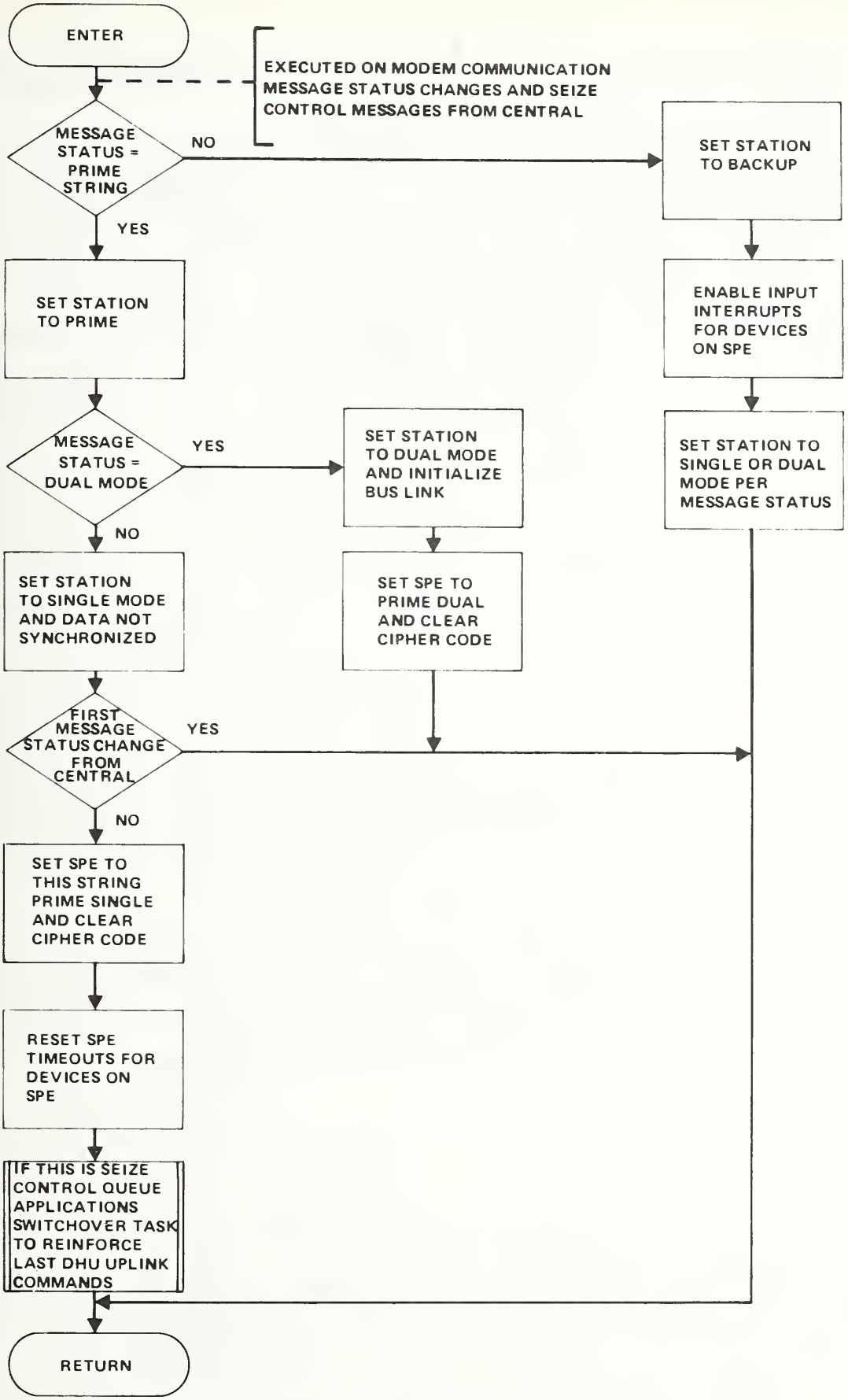

ENTER
EXECUTED ON MODEM COMMUNICATION MESSAGE STATUS CHANGES AND SEIZE CONTROL MESSAGES FROM CENTRAL
MESSAGE STATUS = PRIME STRING
NO
YES
SET STATION TO BACKUP
ENABLE INPUT INTERRUPTS FOR DEVICES ON SPE
SET STATION TO SINGLE OR DUAL MODE PER MESSAGE STATUS
SET STATION TO PRIME
MESSAGE STATUS = DUAL MODE
YES
NO
SET STATION TO DUAL MODE AND INITIALIZE BUS LINK
SET SPE TO PRIME DUAL AND CLEAR CIPHER CODE
SET STATION TO SINGLE MODE AND DATA NOT SYNCHRONIZED
FIRST MESSAGE STATUS CHANGE FROM CENTRAL
YES
NO
SET SPE TO THIS STRING PRIME SINGLE AND CLEAR CIPHER CODE
RESET SPE TIMEOUTS FOR DEVICES ON SPE
IF THIS IS SEIZE CONTROL QUEUE APPLICATIONS SWITCHOVER TASK TO REINFORCE LAST DHU UPLINK COMMANDS
RETURN

FIGURE 3-37. STATION RECONFIGURATION PROCESSING

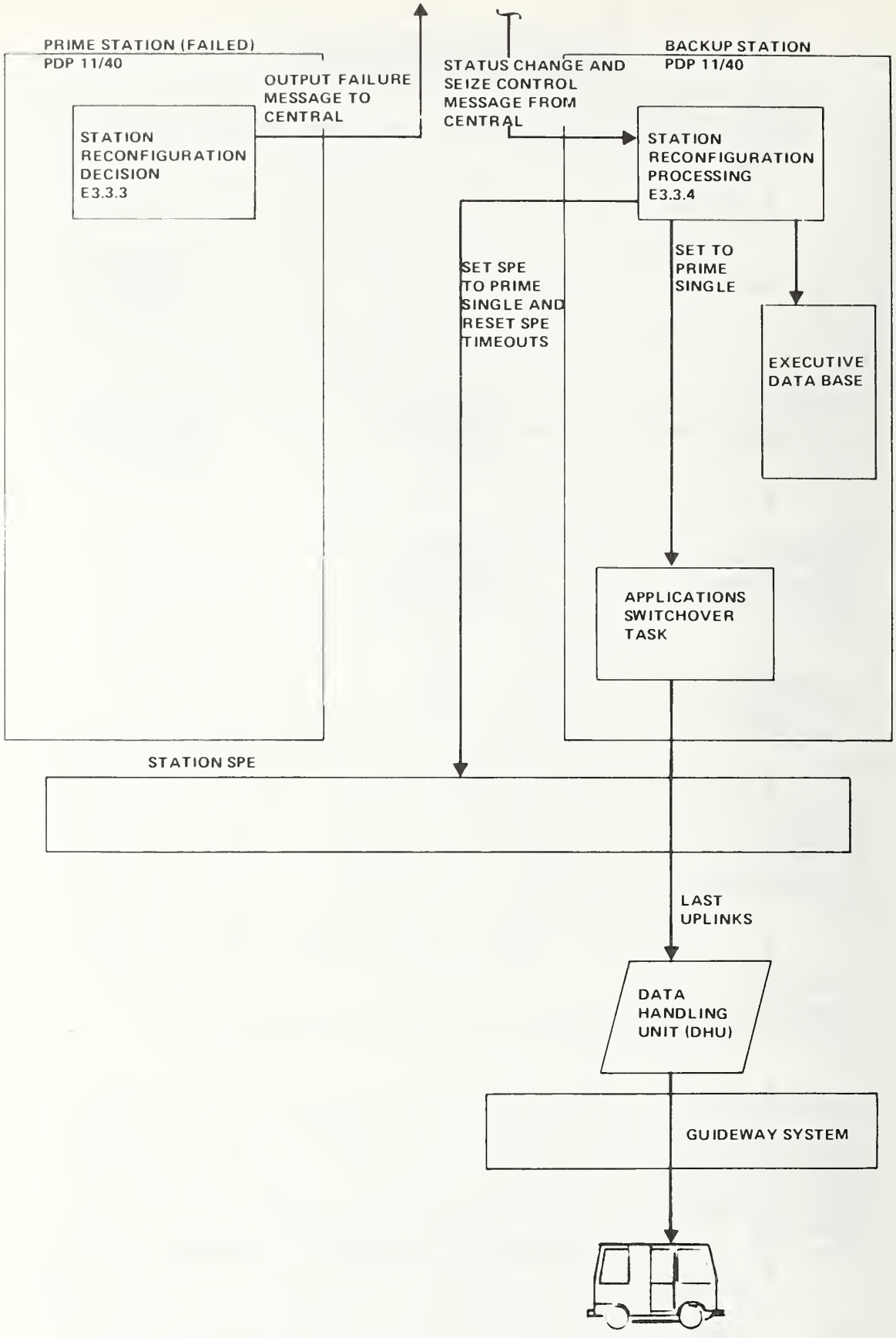


FIGURE 3-38. STATION RECONFIGURATION DECISION AND PROCESSING CONTROL/DATA FLOW

3.4.1 Hardware Components

In the design of the hardware system required to provide a dual string redundancy operation, off-the-shelf equipment was used to the greatest extent possible. This was accomplished at three levels. First various computer systems were examined and a commercial computer family, the DEC PDP11, was chosen. Then various DEC subsystems, such as the DR11-C general interface device controllers, were added. Finally, a design specification was prepared to satisfy the remaining requirements. Even in this specially designed equipment, off-the-shelf modules, backplane building blocks, cables, power supplies, and cabinets were used where possible. As a result, SPE contains about twenty standard module types and has only three specially designed modules required in its fabrication. The remainder of the fabrication is the unique backplane wiring to implement this design.

The favorable aspects for implementation of a system under this approach are reductions in the number of required drawings, in the amount of necessary testing, and in the number of specially designed parts. One undesirable aspect is that some modules will have more capability, thus more complexity, than is required to accomplish the task. This has some adverse reliability impact.

If the same approach were used with present day technology, more system level options would be available. However, unique design would be required to a greater degree at the board level since many functions which were performed at the module level could now be accomplished by individual components.

3.4.2 Software Components

The entire operational software including the executive was custom built for the MPM system. Approximately 5 percent of the total operational software or 2600 lines of executable code perform the redundancy functions. At the time the MPM operational software was built, an off-the-shelf executive could have been used. However, the applications software

and redundancy software would still have to have been custom built. This is obvious when the redundancy functions such as data synchronization, SPE device handling, reconfiguration control, and other-central monitoring are considered. Had an off-the-shelf executive been used, then the implementation of the redundancy functions would have been accomplished in the applications software and by some modifications to the off-the-shelf executive. The cost of implementing the software redundancy functions would have been as much, if not more than the custom-built approach.

If a redundant computing system of any size or complexity were to be implemented today, the lowest cost approach would be to use an off-the-shelf executive and buy as many of the redundancy components as are available. There are many capable real-time control operating systems available today for commercial computers. Computer vendors which offer off-the-shelf redundant computing system components also offer off-the-shelf software to control these components. There are vendors which offer off-the-shelf dual redundant computing systems ranging from cold, to warm, to hot backup capabilities. The cold backup systems are usually implemented with manual device switches or manually transferable data bases on disk packs. The warm backup systems provide some mechanism to update the backup data base in real or near-real-time with manual or semiautomatic switchover capabilities. The true hot backup systems are few and far between, but some do exit. Another option is to contract vendors or system suppleirs to special build the custom software required to implement the redundancy components.

In general, hot backup redundacy schemes today are really in their infancy in terms of off-the-shelf availability. It must be remembered that in a redundant computing system any custom-designed redundancy hardware is almost certainly going to require custom-built software to control it.

4. ANALYSIS AND TEST RESULTS

This section provides results of operating experience obtained with the MPM redundant computing system both in the software development lab and in the field. It contains results of availability measurements obtained during MPM public passenger service and includes analysis and results of the more interesting tests performed on the computing system, a description of some problems encountered, and solutions implemented system a description of some problems encountered, and solutions implemented to correct those problems.

4.1 MPM Redundant Computing System Operating Results

This section contains a summary of the more interesting results of experience with the MPM redundant computing system. It includes a safety record summary, results of field availability measurements, analysis of the causes of dual string divergence, and the results of switchover timing tests.

Safety Record. In section 3 of this report it was stated that the total MPM system is required to provide a level of safety such that there is no more than one accidental passenger fatality per 28 years of service. It was also stated that the computing system is allocated the vital safety critical requirement of removing guideway power in the event of certain failures. The MPM system has been in public passenger service for four years four months as of February 1980, has carried 6.4 million passengers, and has accumulated 2.5 million vehicle miles. During this service the MPM has not experienced a single passenger related injury.

Measured Phase II Availability. The Phase II MPM system has been in public passenger service since July 1, 1979. During public service all downtime events are recorded and attributed to a subsystem. The computer hardware and software are lumped into a single category. Over the 24 week period from July 1, 1979 to December 15, 1979, the

system was scheduled to operate 1745.45 hours. Although there were failures as predicted in the computer hardware system, the redundant computing scheme kept many of the failures from causing system down-time. However, 20 system down-time events were attributed to the computer hardware/software category and amounted to 4.8 hours of system down-time. (This does not include 5.1 hours of down-time caused by hardware/software design deficiencies which were corrected by improvements during this period.) Availability is defined as the ratio of up-time to total scheduled time. Thus, the availability of the computer hardware and software system using the 4.8 hours of down-time is 0.9973 which exceeds the required 0.9969. Table 4-1 shows a comparison of the required/estimated availability parameters to the measured field values for the given time period.

TABLE 4-1. MPM COMPUTER HARDWARE AND SOFTWARE AVAILABILITY - REQUIRED/ESTIMATED VERSUS MEASURED COMPARISON

Required Availability	Measured
Availability	
0.9969	0.9973
Estimated MTBF	**Measured**
MTBF	
175 Hours	87.3 Hours
Estimated Mean Down-time	**Measured**
Mean Down-time	
0.54 Hours	0.24 Hours

Total Scheduled Operating Hours for 24 Weeks, 7/1/79 through 12/15/79 = 1745.45 Hours

The down-time measurement shown here includes more than that caused by the redundant computing system. It includes down-time from such sources as application software limitations and hardware failures which are not designed to cause automatic reconfigurations; also includes time required to get the entire MPM system back up and operating, not just the time to recover the computing system. Of the 4.8 hours caused by the computer hardware and software, 2.75 hours of down-time was caused by two failures originating in the redundant computing system. One was caused by a loose contact in a station SPE which was corrected by reseating a connector. The other was caused by a computer failure in the prime string while being operated in single mode with no armed backup. The backup was not operating due to a previous computer failure in the backup string. The 2.75 hours of down-time caused by failures in the redundant computing system over the 24 week period yields an availability of 0.9984.

This field data shows that the MPM redundant computing system satisfies its availability requirements providing a highly reliable computing system.

Dual String Divergence. One of the most interesting effects of dual string redundancy which surfaced during the test phase is dual string divergence. As explained previously, the two strings are presented the same inputs simultaneously but process these inputs and make output decisions independently. The prime string outputs control the system, and the SPE discards the backup string outputs. Thus, the two strings run in parallel with the prime string providing the system control. Dual string divergence is a situation in which the processing in the two strings has diverged to the point that the two strings disagree on how well the system is operating or on how the system is to be controlled.

The symptoms of dual string divergence vary. Usually, a divergence surfaces as an anomaly report on one string but not on the other;

sometimes the chronological sequence of anomalies occurring at central differs between the strings causing the anomalies to be assigned different sequence numbers on the central operator CRT display. The effects of dual string divergence on operations are minimal. The central operator simply resynchronizes the backup string which immediately brings the backup into agreement with the prime. The worst possible effect is that the system could switchover to an out-of-synchronization backup system. However, analysis and several years of operation have shown no safety problems and no serious operational problems even if this occurs. In the worst case the operational problems would be manifested as stopped vehicles which could be quickly restarted with no complications.

It is interesting to look at the sources of timing differences between the strings which can cause dual string divergence. The following are the major contributors:

- Modem communication errors
- Asynchronous external devices and non-symmetrical device configurations
- Computer clock phasing
- Computer speed differences.

Modem Communication Errors. The MPM system uses 2400 bit per second modems for communication between central and the stations. The error rate for a single line in one direction is less than one error in 10^6 bits. This could mean as many as one error every 35 seconds in one string. Errors in message transmission cause the messages to be retransmitted. A transmission error in one string but not in the other could cause a central-to-station or station-to-central command to be processed as much as 120 ms apart in the two strings depending on the length of message

blocks and on the message backlog at the time of the bit error. Of all the causes of dual string divergence this is believed to be the major contributor.

Asynchronous External Devices and Non-Symmetrical Device Configuration. Except for input devices on the station SPE the devices on the two strings run independent of each other in an asynchronous manner. The devices run at different speeds and at central some peripheral devices, such as the magnetic tape unit, are manually set only to the prime string. At the stations, prime string output to devices on the SPE have to wait for the external device, such as the DHU. However, on the backup the SPE emulates the external device and supplies a faster throughput of messages than on the prime. Communications between strings cause non-symmetrical code sequences to be executed on the transmitting versus the receiving side. These effects can cause computers in the two strings to be executing different sections of the code. For example, a computer in one string could be processing a device interrupt while the other string computer is executing an application task.

Computer Clock Phasing. Each computer is equipped with a clock which runs off the common AC power line frequency supplying an interrupt rate of 60 times per second. All clocks do not interrupt at the same time since not all are on the same phase of the three phase AC power. Thus, the clock interrupt phasing between strings can differ by 5.6 or 11.1 ms for clocks on different power phases. Even with clocks on the same phase, clock interrupts could differ by several computer instruction times between the two strings due to the component variances in the threshold detection circuits in the individual clocks. The difference in clock interrupt phasing causes tasks which are scheduled for future execution to execute at different times between the two strings. An example of this causing dual string divergence is the case in which a task is scheduled to timeout an event such as a vehicle door closing. If the door is slow in closing but very close to the

timeout time, one string will detect the door as timed out and failed while the other (the slower computer) will detect the door as closed and not failed. This would cause an anomaly on one string but not on the other.

For the MPM system it was not practical to retrofit all computer clocks onto the same phase because of the load balancing required for the uninterruptable power supply system in each station. However, since the clock phasing was a known contributor to dual string divergence, the new stations in Phase II were designed with the clocks on the same phase. Future applications should follow this design practice.

Computer Speed Differences. Computers at a location, even though they are the same model, may run at different speeds. The manufacturer's instruction timing information specifies that machines will not differ from each other by more than 20 percent. Typically, machine speeds do not differ by more than about 5 percent.

MPM Dual String Divergence Impacts. Anyone implementing a dual string redundant system needs to determine in advance what impact the sources of timing differences will have in a particular application. The following discussion of the impact experienced in the MPM system may provide some insight for other applications.

The sources of timing differences cause dual string divergence at the average rate of approximately 8 to 10 times per 13-hour day for a normally loaded MPM Phase II system. Approximately 50 percent of the dual string divergence cases manifest themselves as anomaly sequence number differences on the central operator CRT. This is purely an operator display inconvenience and does not require immediate resynchronization. As explained previously, the difference in sequence numbers is caused by anomalies being received or processed in different chronological sequences in the two central computers. At first glance it is not obvious how the order of processing could differ between two computer

strings receiving essentially the same inputs. However, any of the sources of timing differences discussed can cause one computer to have worked off more tasks than its counterpart. Even if a high priority task becomes due and is processed simultaneously in the two computers, when the computers go back to the lower priority tasks the order in which the tasks have executed is different between the two strings. These processing order differences can cause the chronological sequences and, thus, the assigned sequence numbers to differ between the two strings.

Another 37 percent of the dual string divergence cases identify themselves as extra anomalies on the backup caused by a disagreement on the time a vehicle should be dispatched. The vehicles on the guideway run at 15-second separations and are thus dispatched at modulo 15-second release times. The dispatch times are requested by the station and assigned by central. Modem communication errors can cause a delay in the processing of the dispatch time request in one string resulting in different release times to be assigned in the two strings. If the prime string assigns an earlier dispatch time than the backup, the controlling prime string will dispatch the vehicle and the backup will report the vehicle as "moved early". If the prime assigns a later dispatch time than the backup, the backup will unsuccessfully attempt to dispatch the vehicle and will report the vehicle as "failed to depart berth". At first glance it would seem that elimination of this source of divergence could be accomplished by assigning the dispatch times far enough into the future so that both strings would choose the same dispatch time. However, this does not reduce the divergence; it just moves the decision point and dispatch times into the future.

The remaining 13 percent of the dual string divergence cases surface as extra anomalies on the prime or backup caused by a disparity in the detection of off nominal conditions. These are caused by vehicles operating at the outer limits of their performance bands (i.e., vehicles running right on the boundary of moving too fast or too slowly or closing their doors too slowly).

Switchover Timing. Another interesting test result obtained during the test phase is the measured string switchover time. This is the time from failure in one string until the switching of the station SPEs in the other string. The switchover time was measured with an external timer by creating a failure in a station computer and measuring the time until the other string station set the station SPE. Figure 4-1 shows the functional communications required to accomplish the switchover for this case. The failure is reported to the prime central, then to the backup central. The backup central commands the stations to prime single mode. This test was run several times, and the average measured switchover time was approximately 100 ms.

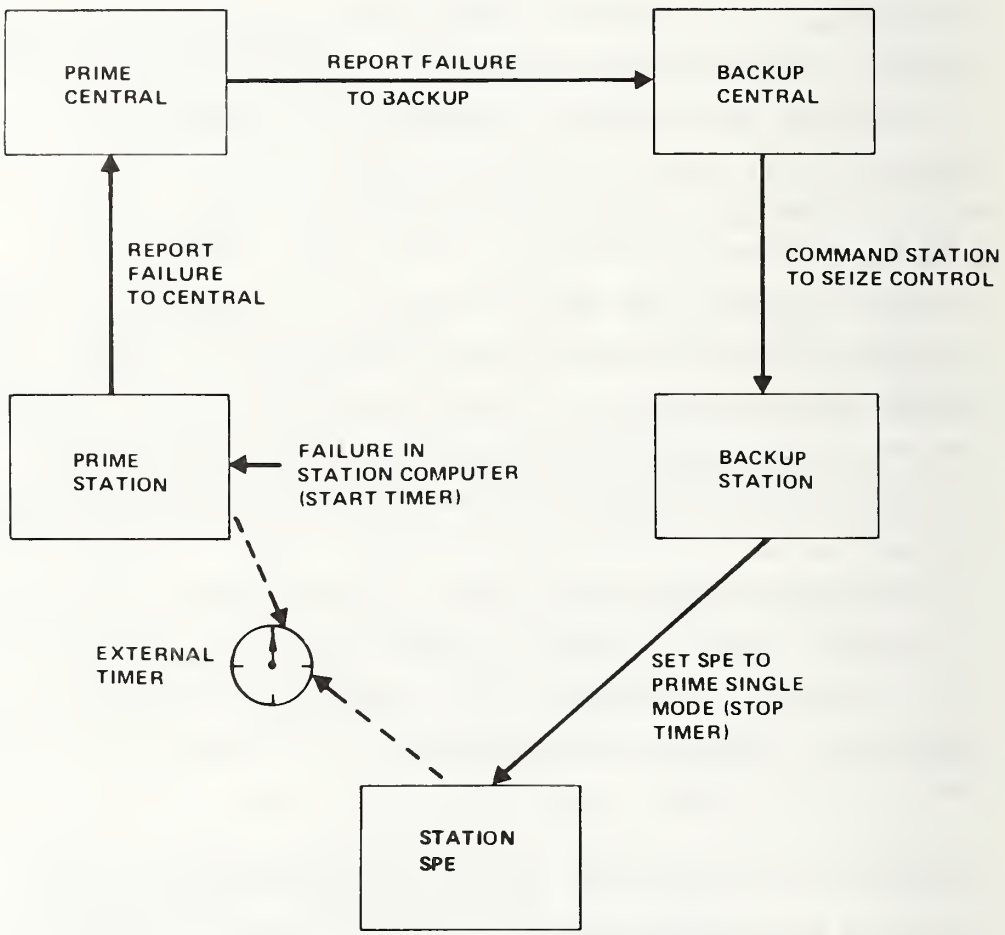


FIGURE 4-1. MEASUREMENT OF SWITCHOVER TIME

Figure 4-2 is a time line showing an estimated breakdown of the 100 ms measured switchover time. Note that most of the time is used by the 2400 bit per second modem communication transmission time between the central and station computers. Figure 4-3 shows this time line elongated for worst case conditions. Note here also that the modem transmission again takes the largest share of the switchover time. The worst case for the modem communication occurs when the messages of interest are backed up behind a maximum sized output block of 36 characters and are contained in a maximum sized block. A 36 character block takes 108 ms (2400 bits per second is 3 ms per character) and two of these blocks require 216 ms to be transmitted. Note that the total of 487 ms is less than the 500 ms switchover timing requirement. This "worst" case is very unlikely to occur because of the probability of occurences of simultaneous worst case events. In fact, a true "worst" case time line would require a more complicated analysis based on probability of events. A more realistic worst case value for central/station transmission time would be closer to 150 ms, for a total of 355 ms used out of the 500 ms maximum switchover timing requirement. If need be, the switchover time could be reduced by decreasing the maximum modem communication block size or by using higher speed communication lines. However, this has not been necessary in the MPM redundant computing system. In the several years of operation the switchovers have never exceeded the 500 ms requirement.

4.2 Assessment of MPM Redundant Computing System

This section contains an assessment of the MPM redundant computing system. It focuses specifically on the merits and unfavorable attributes of the MPM redundancy approach. This section is divided into three areas: general system features, hardware features, and software features.

Overall, the MPM redundant computing system approach has provided an excellent control system. All the system requirements have been met, and the system has proven its ability to meet these requirements over several years of operation.

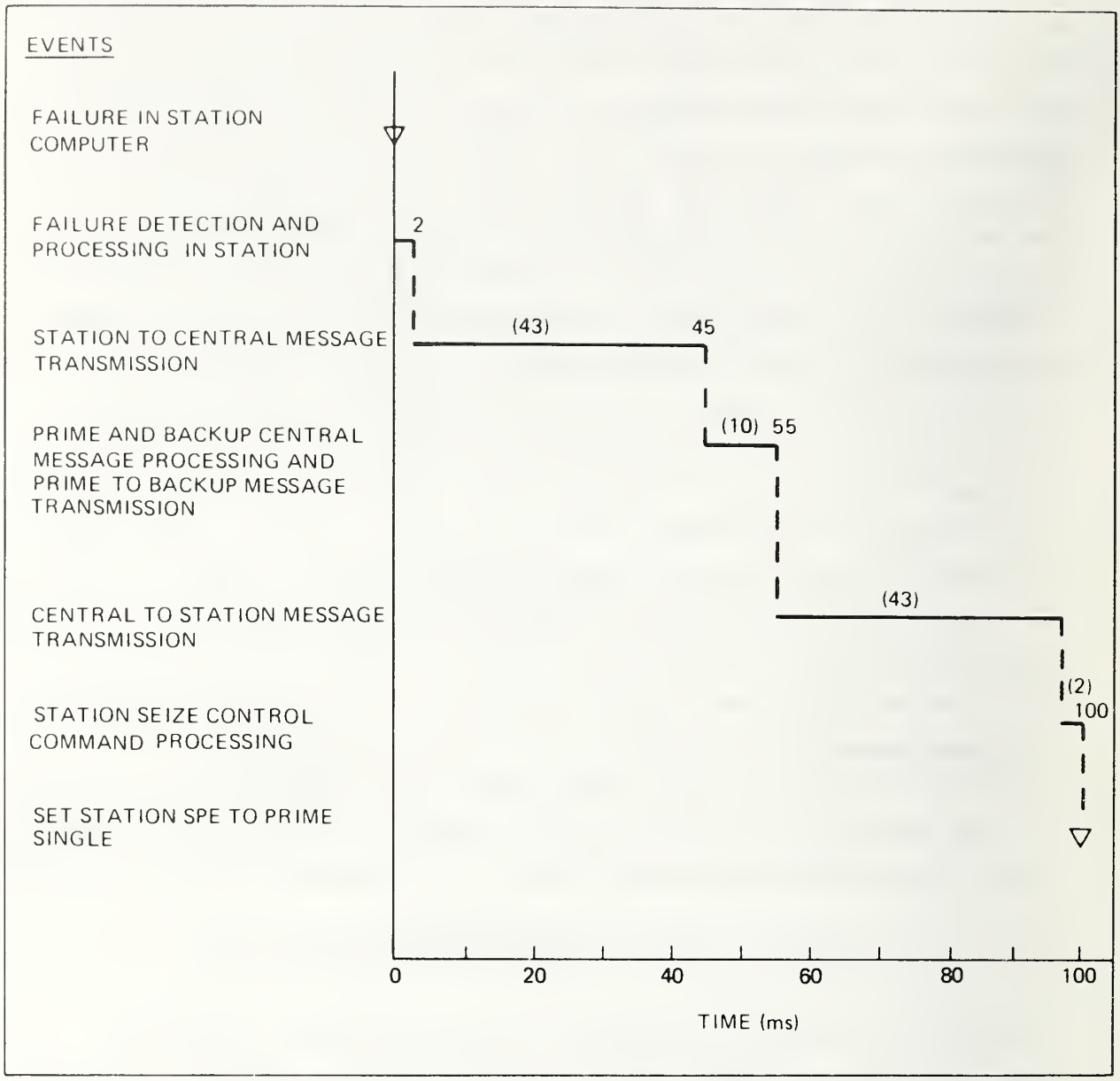


FIGURE 4-2. SWITCHOVER TIME LINE-BREAKDOWN OF MEASURED TIME

General System Features. The single most important advantageous feature of the MPM redundant computing system is that it is fail-operational. A single point failure in the computing system does not cause service interruption. The hot backup and automatic switchover capabilities provide this fail-operational attribute in a distinguished manner. Another important feature of the hot backup system is that the backup is always being exercised and is in fact running in parallel, performing the same operations as the prime string. Also, since the two strings

are identical and either can be prime, the prime responsibility can be switched between the strings periodically, exercising and proving the control capabilities of each string. These two features greatly increase the probability that the failures in the backup system will be detected and corrected before the backup system is required as prime.

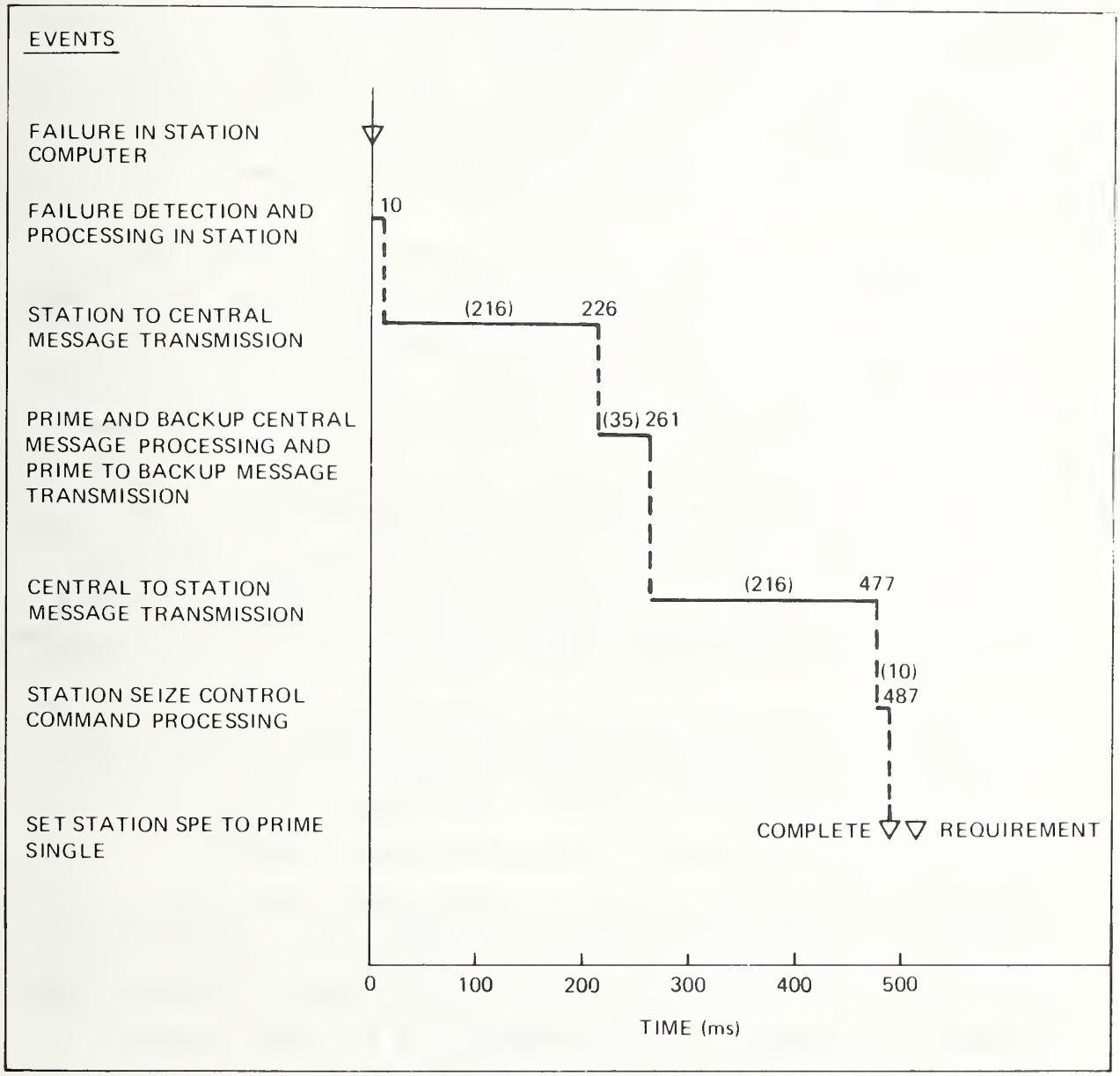


FIGURE 4-3. SWITCHOVER TIME LINE-WORST CASE ANALYSIS

The single most unfavorable attribute of the MPM redundancy scheme is that there is a non-zero probability that a failure will occur in the prime before a previous failure in the backup can be repaired, or that a single point failure in the SPE which may cause down-time will occur. The probability of these events is small enough to be acceptable, and the operational data supports the probability analysis. However, any system down-time due to the computer system is undesirable because it could add to public resistance or resentment of computer based systems of any kind. Another area of concern is dual string divergence. The divergence has proved not to be a problem for the MPM system. However, other applications of this redundancy scheme should keep a watchful eye on this attribute and follow the measures outlined in this report to keep the divergence to a minimum.

Hardware Features. The most important advantageous features of the hardware used in the MPM redundant computing system are that the required availability was achieved using commercial computing equipment and that the special design equipment was kept to a minimum. Commercial equipment has shorter delivery lead times and a much larger user base than military specification equipment, thus, yielding more thoroughly tested products. Commercial products are in production longer and have better vendor support over a longer period of time. Commercial and off-the-shelf equipment do not require special maintenance training and are often easier to maintain than military or specially built hardware.

The central modem reconfiguration unit, which allows manual switching of station computers between strings at a given station, provides an important capability for little cost. In the Phase I system, before this hardware existed, failure of two station computers in opposite strings meant the system was down until one of the two station computers could be repaired. However, this reconfiguration hardware allows for manual switching of station computers between strings to assemble a string of unfailed computers. In the Phase II system the only time the system is down is when both computers at the same location are failed at the same time. This has a significant impact on the system failure and availability analysis. The modem reconfiguration hardware also provides a powerful fault isolation capability on the

station level. For example, faults which switch strings when a certain pair of station computers switch strings can be associated with the switched station computer hardware.

The fact that the two strings are composed of identical hardware has some interesting maintenance and failure isolation benefits. Maintenance personnel have an entire spare computer placed next to the computer they are trouble-shooting. Hard to find failures can be isolated by swapping cards between the two computer strings and observing the results of diagnostic programs. Also, intermittent failures which have long periods, say twenty four hours, or failures which cannot be reproduced using diagnostic programs can be isolated by swapping cards between the strings and observing behavior during system service. Also, since either string can be prime, clues for failure isolation can be obtained by correlating failures with that string which is prime. Failures which occur only when a particular string is prime point to a failure in the one computer string. Failures which occur when either string is prime point to failures other than dual computer hardware.

The ability to isolate a computer string has proved invaluable for maintenance purposes. While one string runs the system, computers in the other string can be powered down, cards can be removed or replaced, and the string can be subjected to diagnostic testing without interfering with the controlling string. Mean-time-to-repair times would be far too large if repairs and troubleshooting could only be performed during nonservice periods. This isolation capability is not a natural attribute of redundant systems. It requires a conscious requirement and design effort to achieve.

Any availability-critical control system must have a fast load capability in order to keep the mean time to repair to a minimum. In the case of distributed systems this can take the form of high speed communications or storage throughout the distributed system. In the MPM system, floppy disk storage at each station supports the fast load capability. This capability not only reduces the mean time to repair but provides substantial time and thus cost savings during the software development and test

phase when each test requires at least a partial system load. The fast load capability reduced single string load time from 13 minutes to 1 minute 15 seconds. The test phase time savings amounted to this: 11 minutes 45 seconds per load times 10 to 15 loads per day times 6 months of software testing. The floppy disks at the stations also provide an excellent medium for loading diagnostic and preventative maintenance software. From the maintenance standpoint the floppy disks are better than the high speed communications from central because troubleshooting or preventative maintenance would require central support if the diagnostics are down loaded from the central facility.

On the adverse side, experience has shown certain features of the MPM redundant hardware which are undesirable. These features create minor problems which have not been eliminated because modifications could not be justified on a retrofit basis. However, if a redundant system similar to the MPM system were to be built, it would be well to avoid them. The MPM system has experienced some problems with the operators' being confused by the configuration of the auto boot panel used to initiate computer loading. The panel shown in Figure 4-4 is mounted on one of the A string central computer racks. The panel consists of two rows of push buttons marked "A" and "B" string with turn keys to arm or disarm the pushbuttons for each string. The confusion arises when the operator tries to load the backup system but activates the boot strap loaders for the wrong string. This is a human engineering problem and could probably be solved by simply splitting the panel into two panels with the A boot strap loader buttons in the A computer racks and the B in the B racks thereby physically locating the pushbuttons with the string they control.

Another desirable feature would be to make the guideway power trip interface at central more fail-safe. Currently, the software in either string can actively trip portions or all guideway power. In conjunction with this capability the power trip should be a "hold down" interface so that the software must periodically reinforce the power-on status or the power would be automatically removed. This would automatically

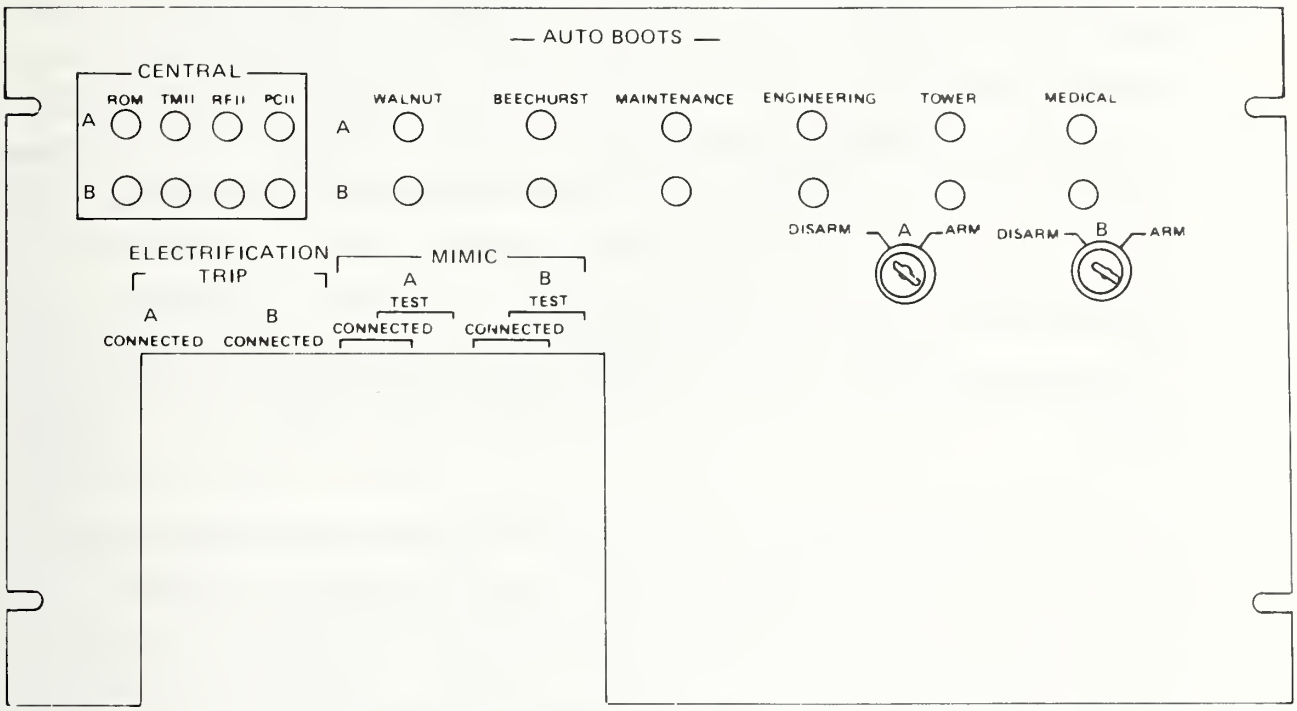


FIGURE 4-4. MPM REDUNDANT STRING AUTO BOOT PANEL

remove power in case the prime central computer with no armed backup seizes up and is unable to trip power due to its own failure. As in the current system, the operator would still have to provide a manual power removal backup in case of certain or multiple hardware failures, but with the "hold down" power interface the operator backup would not be needed as often.

Software Features. As explained previously the software is divided into the executive and applications software with the redundancy functions assigned to the executive. The fact that the applications are nearly blind to the existence of a second string greatly simplified the software development and test phase. Development and product assurance test cases which test application functions need only be run on the prime

string. If there were any prime/backup decisions in the application software, the number of test cases and, hence, the testing effort could have easily doubled. The same argument holds for using the exact same software image on the A and B strings with no software decisions based on whether the software is residing in the A or B string. If the software for A or B strings differed, or if the software made A or B string decisions, the test effort would have been much larger. This also aids in isolating system bugs and component failures. For example, anomalies which occur only when a particular string is prime cannot be software problems since the software has no knowledge of A or B status.

One of the unique features of the software system is the custom built executive. An off-the-shelf operating system was used for software development to build and maintain the operational software utilizing off-the-shelf assemblers, compilers, file control systems, and other software development facilities. However, the operational software real-time control executive used for operating the MPM system was custom built. Building the executive software instead of using an off-the-shelf executive was justified based on unique MPM requirements, ease of verifying subcontracted applications software, the requirement for a common applications/executive interface in all computers in the network, capability, efficiency, and, of course, cost and risk. The redundancy management, data synchronization, system status monitoring, and configuration control functions needed to be developed regardless of the executive used; therefore, the extra cost for the custom built executive was not as large as one would expect.

The MPM executive has proved to be an excellent real-time control executive. It is extremely reliable, very efficient, and relatively simple and small because it is tailored to the capabilities required for the MPM system. Since it is small and entirely under control of the company performing the system development, test, and final system integration, it is easily modified. Off-the-shelf or vendor supplied executives or operating systems are large by nature, built to satisfy a wide range

of requirements, and hopefully supply capabilities to satisfy all user's needs. Modifying an off-the-shelf executive to change or add a capability or to solve a problem can be complicated and risky for customer organizations. Also, getting a vendor to supply a modification for a development schedule critical project or to solve a field problem which is causing down time for an availability critical control system can be costly and frustrating.

The custom built executive provided excellent flexibility and design latitude during the software requirements, design, and implementation phases. The custom executive was easily tailored to provide the redundancy functions since the design of the redundancy functions and allowances for the dual string architecture were made in parallel with the design of the rest of the executive. The Data Synchronization and Reconfiguration Control modules would have been much more difficult to design and implement under the constraints of an off-the-shelf executive. The requirements on the Data Synchronization module, to allow data synchronization at any time on the fly with no interruption in passenger service, and the approach used, to temporarily suppress all device input/output to accomplish almost total variable data synchronization, would have been very difficult to achieve in interfacing with an off-the-shelf executive. These requirements would have almost certainly dictated very difficult user modifications to such an executive.

One of the custom executive draw backs which has surfaced is that several of the peripheral devices the executive utilizes have become obsolete and unavailable for reorder. Since the new devices are not software compatible, substitution of a new device requires a new executive driver. This is an advantage of off-the-shelf executives. New devices are often accompanied by vendor supplied device drivers for off-the-shelf executives reducing the cost to replace obsolete hardware and allowing the buyer to upgrade to better hardware devices on subsequent deployments of the product. Also, off-the-shelf executives allow purchase of application software compatible, upgraded executives at a cheaper cost than those specifically built since the cost of the upgrade is amortized over many users.

As explained previously, the MPM executive is small and efficient. This is an advantage over off-the-shelf operating systems which by nature require more computer hardware resources (memory and speed) to compensate for inefficiencies inherent in general purpose operating systems.

In retrospect, the custom built executive for the MPM redundant computing system has proved its worth in capabilities, efficiency, simplicity, flexibility, and ease of use and modification but probably costs more than purchasing a modern off-the-shelf executive and developing only the redundancy functions of the executive in house. Redundant computing systems in the medium to large size range developed in the future range should use an off-the-shelf operating system if at all possible.

4.3 Problems Encountered and Solutions Implemented

This section discusses some of the interesting problems which occurred over the history of the MPM redundant computing system and the solutions implemented to correct these problems.

Central-Central Communications. Midway through the Phase IB project the central bus link used for central-to-central communication proved to be somewhat of an Achilles' heel. When this device failed, each central timed out the other and seized control in prime single mode. The seizing control was a race during which both strings set the station SPEs to its own string in prime single mode. This usually resulted in some station SPEs set to one string and some to the other and neither string receiving enough system data to control the entire system. This case of a single point device failure causing system failure was not acceptable. A secondary central-to-central communication loop which consisted of just three read/write bits cross wired between the two central computers was added. This provided enough communication capability for the two centrals to arbitrate other-central timeouts. Central-to-central bus link failures now result in one string seizing control and the other string yielding with no system down time.

Prime/Backup Error Encoding. Failures which occur throughout the computing system are reported to Central Reconfiguration Decision for failure reaction. When the software Reconfiguraton Control package was first delivered to the field for Phase I installation and checkout, software detected failures were encoded with prime or backup status at the point of failure detection. Central Reconfiguration Decision compared the failure message prime/backup encoding to the prime/backup status of the string it was residing in to make reconfiguration decisions. This resulted in confusion in the software since prime/backup status changes during switchovers. An example of this problem is when a backup station detects a failure in the prime string, such as a SPE timeout. The station reports the failure as a prime string failure. Central commands a switchover to the backup string and changes from backup to prime. If station reports another prime (other) string failure before the station learns it is now the prime string, central (the new prime central) will interpret the second failure as a failure of its own string. The solution to this problem was to change the sense of the prime/backup encoding to "this string" or "other string". This eliminated the confusion since this/other string status does not change during switchovers and multiple failures.

Dual String Divergence. One of the problems which caused concern through the Phase I and during the design phase for Phase II was the number of occurrences of dual string divergence. The reason for the concern was the operator work load to resynchronize the backup and, since the backup is disarmed during the synchronization process, the exposure to a single point failure in the prime causing system down time. In a normally loaded Phase I system the occurrence of dual string divergencies averaged approximately 10 per 13 hour day and ranged up to a maximum of 21 per day. Projections for Phase II predicted an average of 19 per day. The Phase II prediction was not large enough to initiate a problem solving effort in itself. However, results of Phase II software development tests showed that for central/engineering communications alone, a data transmission error was occurring every 6 seconds compared with 18 to 25 seconds for the Phase I system. Since the modem communication error rate was a known contributor to dual string divergence, the increase

in the error rate warranted further investigation. Analysis showed that the communication errors were induced by data overruns during the input interrupt processing of PD data at a higher priority level than the modem character interrupt level. This means that the processing of PD inputs was locking out the processing of modem characters causing communication errors.

Analysis showed that these induced errors could be eliminated with no ill effects by interchanging the hardware interrupt priority levels of the two devices, making the modem interrupts a higher priority level than the PD interrupts. Tests with the software development lab configured to this change showed the previous error rate for one central/station pair of one error per 6 seconds drop to zero for a 10 minute run. This reduction in modem communication errors is the largest contributor to the reduction in dual string divergency occurrences from the predicted average of 19 per day to an average of approximately 8 to 10 per day for the Phase II system. This level of dual string divergence is acceptable and has not caused any operational problems.

Guideway Power Removal By Backup String. Another problem caused by dual string divergence was guideway power removal by an out of synchronization backup string. Since only the prime string controls the system, there are conditions in which the backup string goes far enough out of synchronization that it thinks it detects an unsafe condition requiring guideway power removal. An example of this is when the prime dispatches a vehicle that the backup believes should not go until the next 15 second time slot on the guideway. The backup application software outputs commands to stop the vehicle and check the integrity of vehicle to computer system communications. Since the station SPE inhibits output of the backup string vehicle message, the backup string does not receive its expected responses and the communication test fails. The backup string believes it has detected a communication failure and trips guideway power in the section where the vehicle is located.

One part of the solution for this problem was to prevent the executive from ever allowing the backup applications software to trip power. Thus, anomalous conditions detected by the backup due to dual string

divergence can no longer cause power removals or vehicle stoppages. Another part of the solution was for the central application software to reissue the current power trip status on switchovers to cover any possible timing holes since backup outputs for power removal are inhibited.

5. POTENTIAL SYSTEM IMPROVEMENTS

This section describes the limitations of the MPM redundant computing system as it now exists and shows how these limitations can be removed by making improvements. This section also describes the limitations of the MPM redundant computing system concept and what can be done to remove these limitations. The MPM redundant computing system as it now exists has satisfied all its requirements for the MPM system. However, the improvements described in this section could improve the system availability, reliability, and operability, and increase the range of applications for the MPM redundant computing system.

Improvements To Existing System. Using the existing hardware/software configuration the MPM system is limited to the addition of three or four more station computers. The limiting factor is the central CPU loading. Figure 5-1 shows the CPU utilization for the central computers as a function of number of station computers and number of vehicle dispatches per 5 minutes per station. The number of vehicle dispatches per 5 minutes is a measure of system activity with more dispatches indicating more activity. The Phase II system of 6 stations at its maximum loading of 60 dispatches per 5 minutes has an average CPU utilization of approximately 48 percent for the central CPU. The increase in utilization is approximately 8.2 percent per added station with 10 added dispatches per 5 minutes. Approximately 4.6 percent increase in total CPU utilization per station is due to the central/station communications.

This limitation could be most effectively removed by replacing the DP11 serial synchronous line interfaces with modern interfaces that reduce the CPU utilization requirements. The DP11 interfaces, which were the only communications option available at the time the hardware configuration was chosen, require the CPU to input/output each eight bit character and perform all the line protocol including error checking and recovery. There are communication interfaces now available which perform all the line protocol, error checking, error correcting, and only call on the CPU when a block of messages is ready for processing.

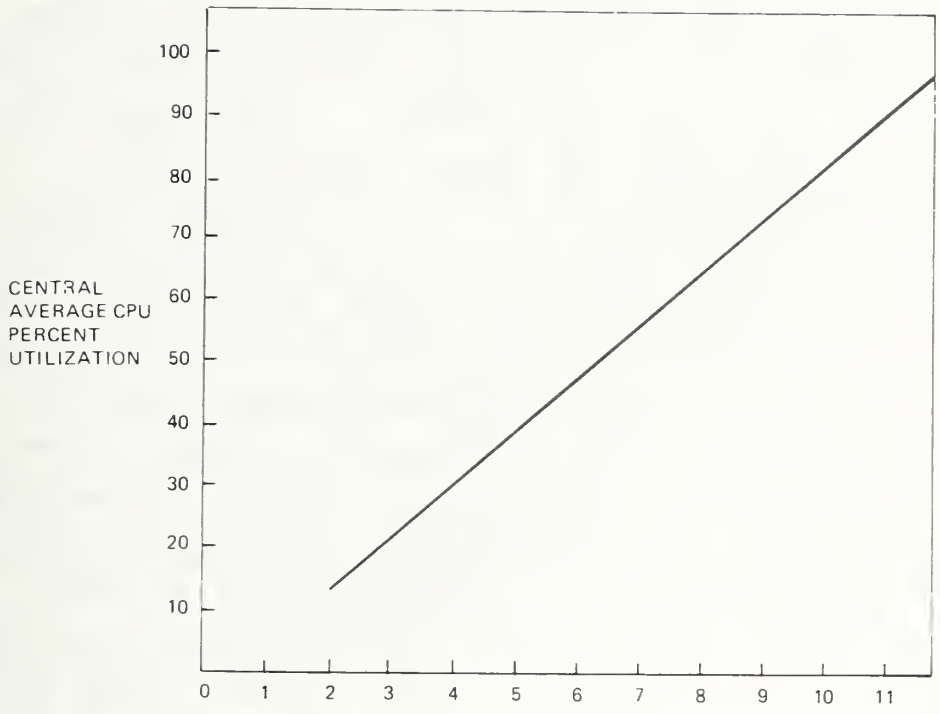


FIGURE 5-1. CURRENT CENTRAL AVERAGE CPU UTILIZATION

If these newer communications options would reduce the central CPU utilization from 4.6 percent to say 2 percent per additional station the MPM computing system could be expanded to fifteen or sixteen stations as shown in Figure 5-2.

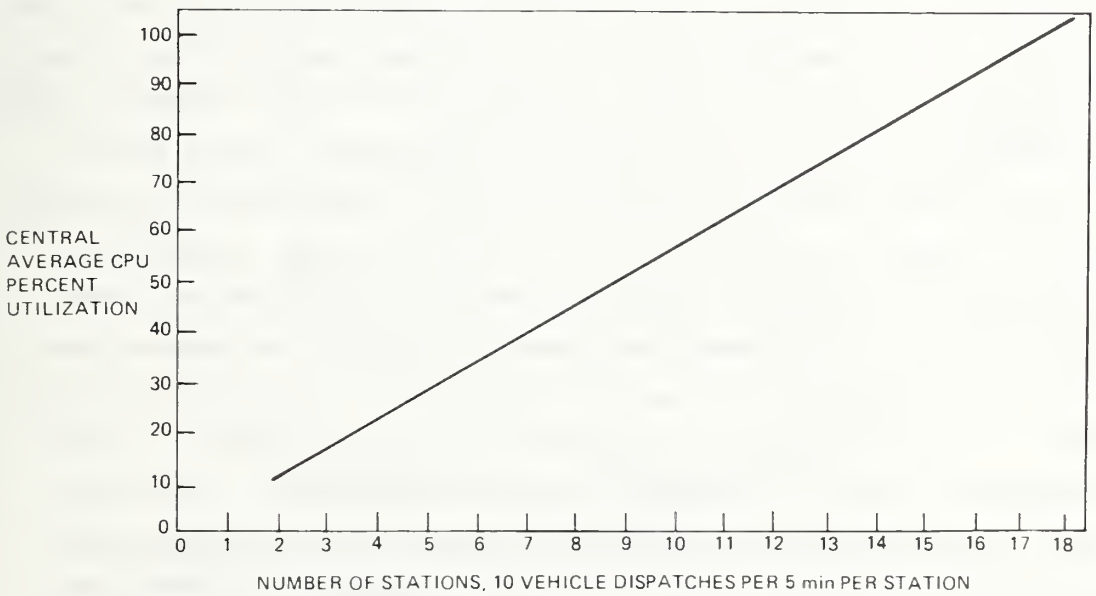


FIGURE 5-2. PROJECTED CENTRAL AVERAGE CPU UTILIZATION WITH IMPROVED COMMUNICATION INTERFACES

Another limitation of the system as it now exists is that a failure of a station computer with no backup string requires the entire system to be shut down. The system remains down until an entire good string is assembled using the modem reconfiguration unit and restarts or until the system is restarted with the rest of the operable stations. An improvement which would remove this limitation would be to shut down only the station which failed and to allow the rest of the system to operate uninterrupted. If the system then allowed a single station load and restart while the rest of the system operated, the station computer from the backup string could be manually switched in using the modem reconfiguration unit and restored to operation without affecting the rest of the system. This improvement was considered during the Phase II requirements phase. It was especially attractive because it was a software only change. However, it was determined by analysis that the required system availability could be achieved without the cost of this improvement, and this analysis has been supported by operational experience.

Extension of MPM Redundant Computing System Concept. A limitation of the MPM redundant computing system concept is the approach of switching in the entire backup string instead of just one station from the backup string. A higher level of availability could be achieved with a station by station switch into the prime string on failure of the prime string stations. The current MPM system allows an individual station to be switched in but only on a manual basis when the station computers which are switching strings are not operating the system. The only time a single station switching becomes desirable is when the entire backup cannot be loaded to provide the full string switchover capability. The hardware to accomplish this improvement was developed and used in the MPM Phase II software development facility. The hardware was similar to the modem reconfiguration unit, but it provided for software control of the switching of communication line connections. In the development lab any two computers could be connected to talk to each other and a third could eavesdrop on the communications of any two. In the lab these capabilities were used to automatically configure the five test computers for a software test configuration saving considerable

manual test setup time. In the operational system these capabilities could be used to allow automatic on-the-fly switchin of a backup station computer. The backup string would run in parallel with the prime as in the current system but could consist of less than the full set of backup stations. The backup central would eavesdrop on the communications of the prime station or stations it is missing from the backup string. Figure 5-3 shows functionally how the backup central eavesdrop would be accomplished using the modem reconfiguration unit modified to allow this capability. The eavesdropping would compensate for the missing stations and allow the backup string to stay in synchronization with the prime. The backup stations would be available to replace the prime stations on the fly as needed. When a backup station is switched in it would have to establish communications with the prime central. This would amount to some reinitialization of the modem communication line protocol. A special "retart communication sequence" command from central could accomplish this establishing of communications.

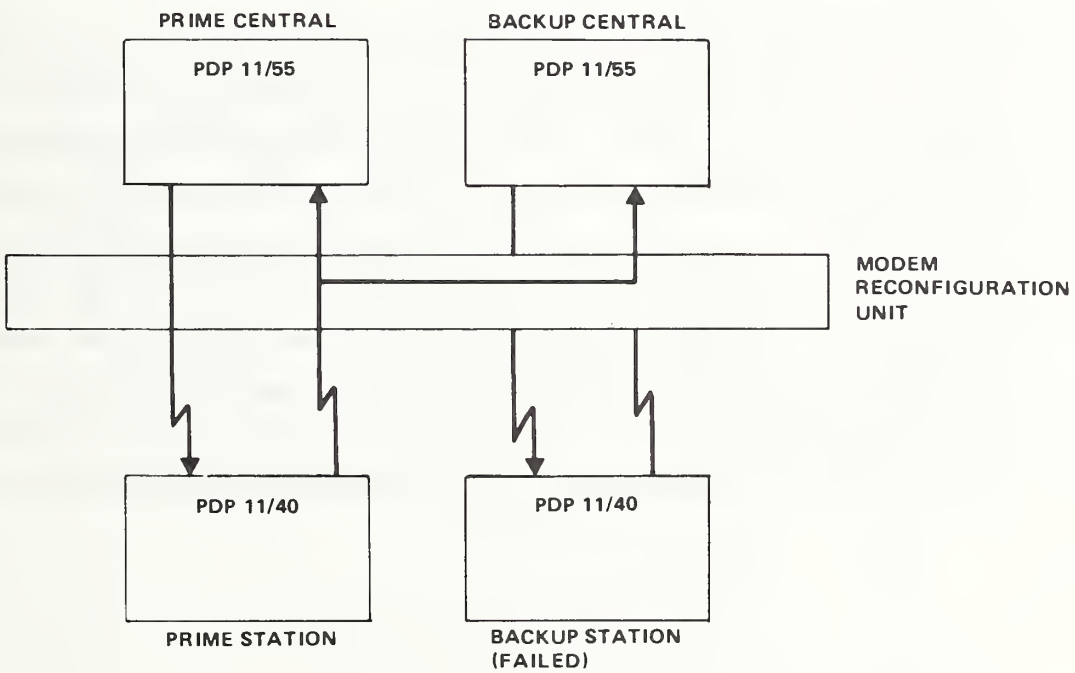


FIGURE 5-3. BACKUP CENTRAL EAVESDROP ON PRIME TO COMPENSATE FOR FAILED STATION IN BACKUP STRING

The redundant string concept has an inherent growth limitation because of the number of station computers one central computer can handle. This could be handled in many ways such as by front end communication processors or by hierarchical configuration. However, the method which would cause the least impact to the existing MPM redundant computing system and which would double the size of the fifteen stations achieved by the change in communications interfaces would be to add a second set of two strings to control the additional expansion. The two sets of strings could run as adjacent neighbors with handover from one central to the other similar to the station-to-station handover. Each set of dual strings would control a region. The adjacent centrals could have cross coupled communications as shown functionally in Figure 5-4. This would allow independent prime/backup designations for the two sets of strings. The addition of sets of dual strings could be carried on indefinitely with each set of dual strings controlling a region. Overall supervisory status/control could be provided either by one designated central or by all the centrals reporting to an additional computer. The later configuration is shown in Figure 5-5.

The MPM redundant computing system has satisfied all its requirements in the MPM system and has proved to be a reliable, maintainable, and operable computing system. The redundant computing system developed for the MPM system could be used for a wide range of applications. It is especially suited to real-time control of geographically distributed systems with relatively high response time requirements but could also be used for centralized applications with the same or lower response time requirements. The MPM redundant computing system concept of dual string redundancy, parallel hot backup processing, and real-time switchover with no control interruption has an even wider range of applications.

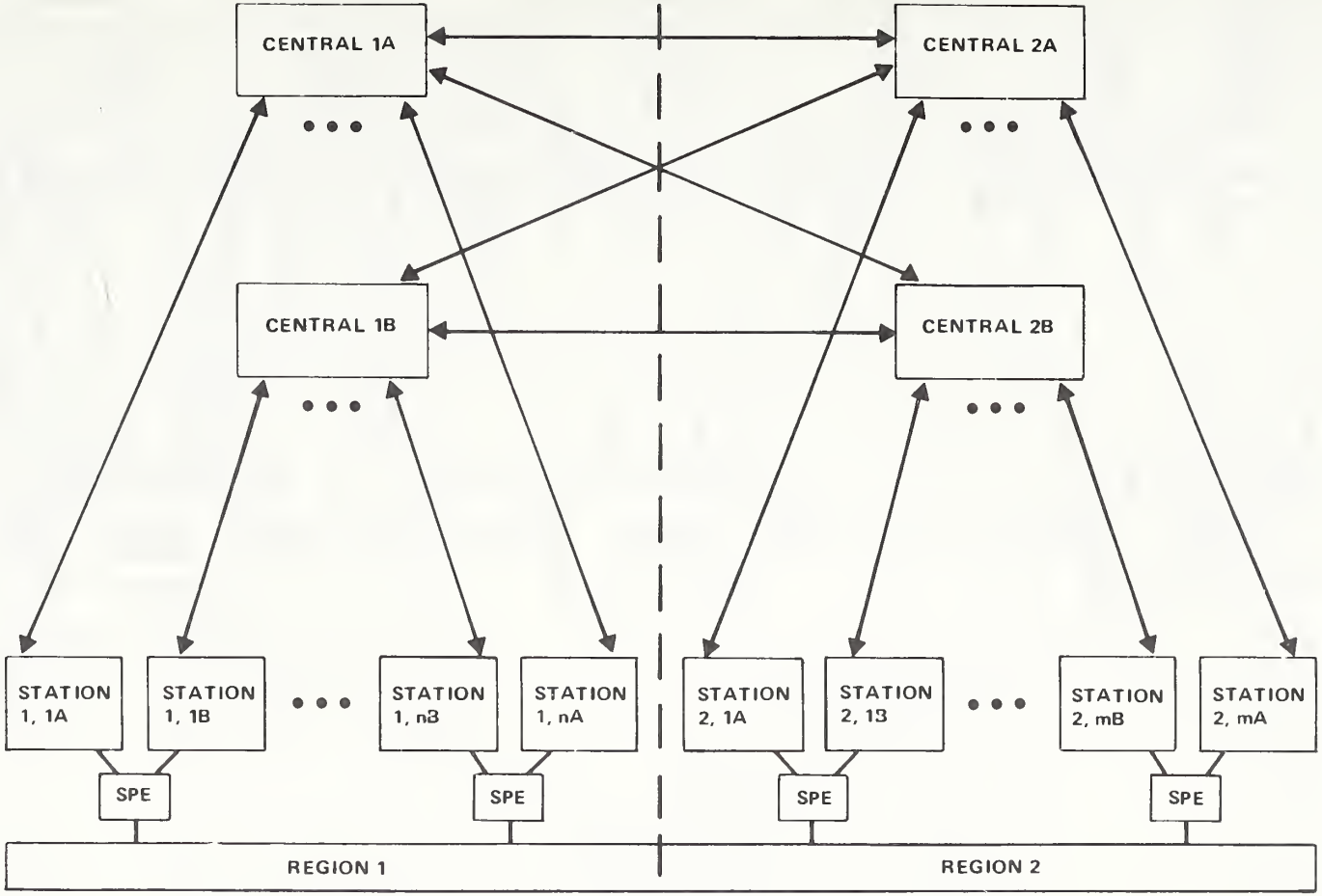


FIGURE 5-4. ADJACENT SETS OF REDUNDANT COMPUTING STRINGS

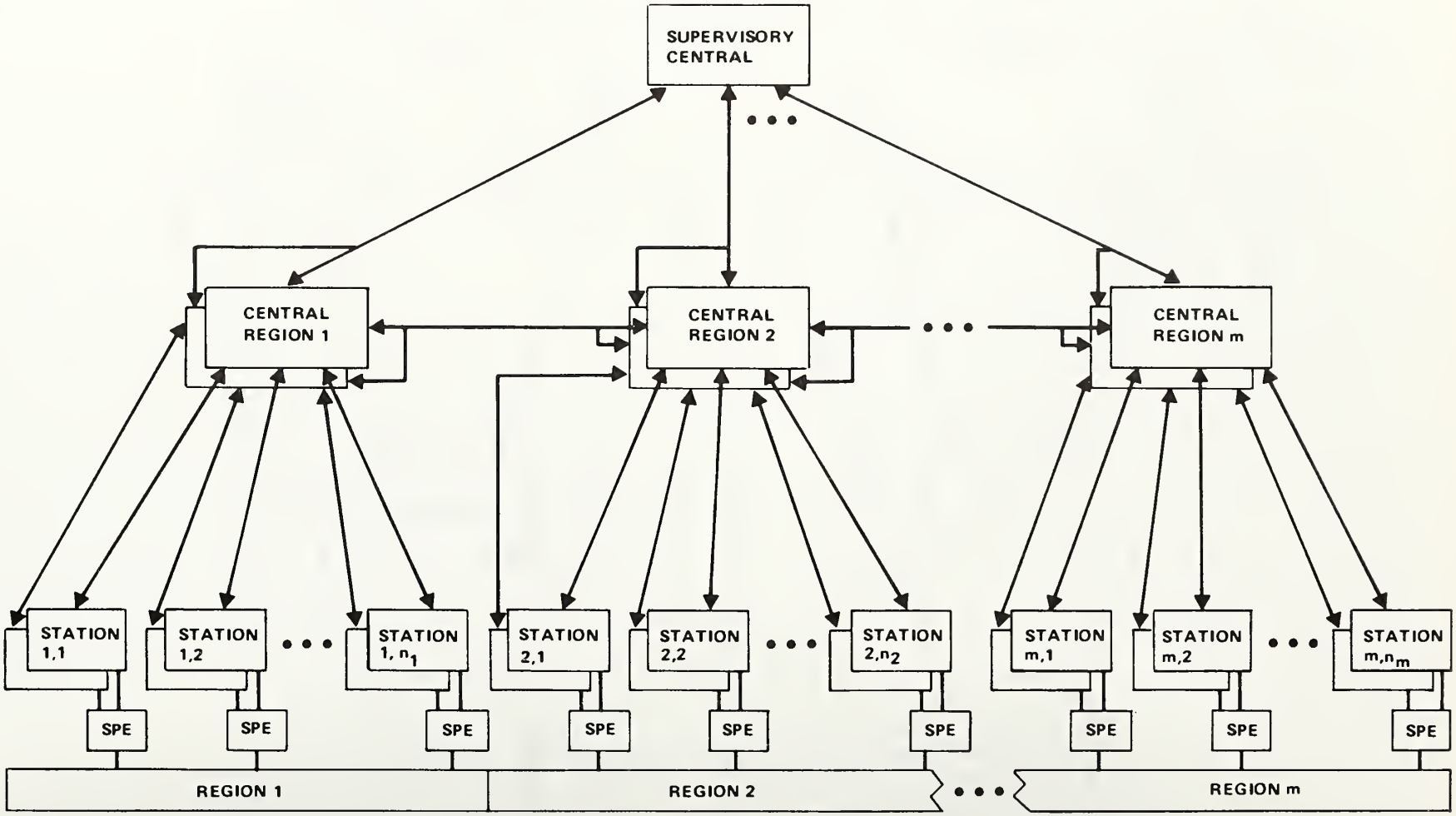


FIGURE 5-5. REGIONS WITH A SUPERVISORY CENTRAL

6. RECOMMENDATIONS

This report summarizes the design of the MPM redundant computing system and represents a considerable amount of experience with the MPM redundancy approach. The following recommendations should provide some guidance for implementing a dual string redundant system whether it be similar to the MPM scheme or to some other approach.

Dual Redundancy Complexity

The user should never underestimate the complexity of making a computer system dual redundant. The complexity of adding a second string spreads into every phase of the project: requirements, design, implementation, development testing, verification testing, system operation, and system maintenance. The ramifications of going to a redundant system also spread into more areas than first realized: maintenance, operations manuals, operator training, and so on. A good guideline to follow is that if the redundancy job looks simple, the situation has not been totally grasped.

Dual String Symmetry

In dual string redundant computing systems, the exact same software should always be used in each string, the hardware computing strings should be identical, and the software should not be allowed to make decisions based on its location in either the "A" or "B" string. Failure to adhere to these rules will lead to a greatly increased test phase since the software must be tested in each string and to the loss of a powerful failure isolation capability due to the loss of symmetry between the strings.

Dual String Divergence. Special attention should be paid to dual string divergence. The sources of divergence in the hardware and software should be reduced whenever possible. Also, the end effects of dual string divergence in the system should be reduced. Examples of both the sources and effects in the MPM are given in the body

of this report. These will provide a guideline to accomplish this end.

SPE Hardware

Special attention should be paid to the requirements, design, and test of any "SPE" hardware inserted at the point the system changes from the dual computing system to the single string portion of the system application. The single to dual transition point is a potentially troublesome area since the ramifications of the transition point are usually not totally understood or provided for on the first attempt. The design of this SPE equipment should be kept as simple as possible and as many requirements as reasonably possible should be allocated to off-the-shelf computer interfaces and system software.

Redundancy Ramifications

The redundancy functions should be restricted to as few subsystems and components as possible. This will reduce the test and development efforts and increase off-the-shelf availability of system components.

String Isolation

In dual string redundant systems, provisions should be made for string isolation for maintenance purposes. Allowance should be made for work on as much of the string which is not operating the system as possible without affecting the controlling string. Allowance should be made for performance of diagnostic programs and troubleshooting precedures, such as powering down of the second string. It is of paramount importance in order to achieve high availability to provide for noninterfering maintenance on one string while the other string operates the system uninterrupted.

Communication Interfaces

In future applications of the MPM redundant computing system the modem communication interfaces should be replaced with modern

higher speed interfaces, at least 7200 bit per second, and with interfaces which do not require the CPU to perform the line protocol, error checking, and error recovery. This will reduce the central CPU utilization allowing more station computers to be added to accomplish system expansion and will reduce dual string divergence. The floppy disk storage at the station, although it would no longer be needed for the fast load capability, should still be provided because of its advantages as a maintenance diagnostic system storage medium.

APPENDIX A - GLOSSARY

Application Programs -	Those computer programs which perform functions pertaining directly to the end problem being solved or application being implemented. In the MPM system, those programs which control system operation from the passenger and operator point of view.
Arm -	To make the backup computer string eligible to seize control and become the prime single string to control the system.
Availability -	Probability that a system is ready for use at a random point in time. Also, the ratio of system uptime to total scheduled time.
Backup String -	The combination (string) of computing equipment which runs in parallel with the prime string but not controlling system operations.
CAP -	Central Applications Program

CAS -	Collision Avoidance System
CCCS -	Central Control and Communications Subsystem
C&CS -	Control and Communications System
Collision Avoidance System -	A system whose fail-safe design operates to prevent vehicle collisions. The CAS is designed so that a vehicle approaching an occupied CAS block will lose its safetone in the block ahead. Loss of safetone results in a braking rate which will prevent entry to the occupied block.
CRT -	Cathode Ray Tube
Data Acquisition Unit -	That part of the station electronics which provides guideway input data such as presence detector data to the computer system.
Data Handling Unit -	That part of the station electronics which provides vehicle to computer system message input (downlink), and computer system to vehicle message output (uplinks).

Data Synchronization -	Synchronize the operating state of the backup computing string to that of the prime string by transferring the variable data base contents of each prime string computer to each corresponding backup string computer.
DAU -	Data Acquisition Unit
DEC -	Digital Equipment Corporation
DHU -	Data Handling Unit
DMA -	Direct Memory Access
Downlink -	A message from a vehicle to the station computer
DSU -	Destination Selection Unit
ESR -	Executive Service Request
Executive Software -	The computer software which acts as the software operating system. In the MPM system, the executive controls the processing performed by the applications programs and provides the softwareinterface with the computing system and external environment.

FSK - Frequency Shift Key. This type of modulation is used for the MPM uplink/downlink communications.

Guideway - A dedicated roadway along which vehicles are guided.

JPL - Jet Propulsion Laboratory

Mimic Board - A graphic display in the central control room which shows guideway layout, vehicle locations, and the existence of anomalous conditions.

Modem - Modulator/demodulator device for digital communication used for central to station and station to central computer communication in the MPM system.

Modem Reconfiguration Unit - A hardware device in central which allows station computers to be switched between computer strings.

MPM - Morgantown People Mover (formerly MPRT - Morgantown Personal Rapid Transit)

ms - Millisecond

MSAP - Maintenance Station Application Program

MTBF - Mean Time Between Failure

Operating System - A computer software system under which software programs can be executed and often includes software development capabilities.

Passenger Boarding Display - A computer controlled lighted display at the stations which aid passengers boarding the vehicles.

PBD - Passenger Boarding Display

PD - Presence Detector

Presence Detector(s) - Magnetic reed switches which are mounted at fixed positions along the guideway (to provide vehicle location data to the control system). PD's are activated by magnets mounted on the underside of each vehicle.

Prime String - The combination (string) of computing equipment which controls system operations.

PSAP - Passenger Station Application Program

Safetone - An inductive communication signal to the vehicle indicating that it is safe to proceed through a CAS block.

SAP - Station Applications Program (refers to PSAP or MSAP)

SCCS - Station Control and Communication Subsystem

SPE - Special Purpose Equipment

UMTA - Urban Mass Transportation Administration

Uplink - A message from the station computer to the guideway and/or vehicle.

VCCS - Vehicle Control and Communications Subsystem

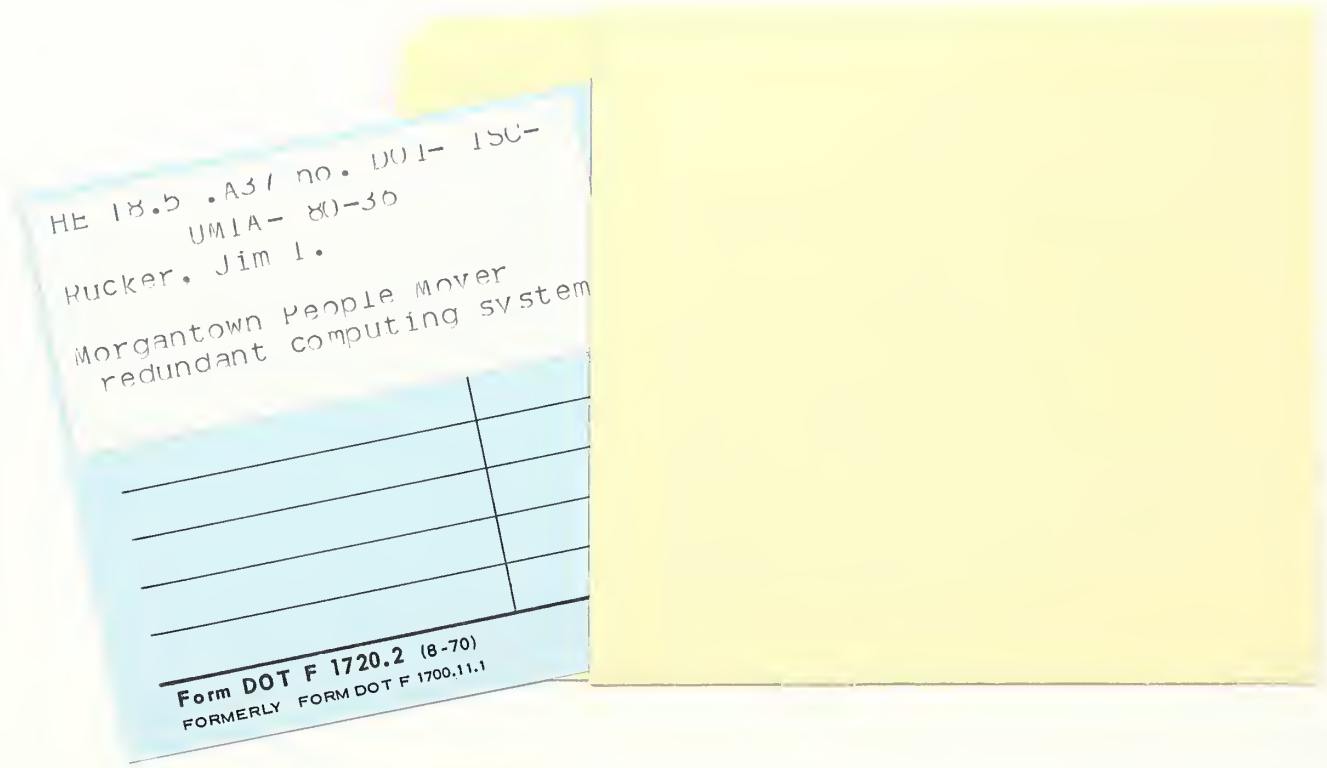
Virtual Point - A set of mathematical points maintained by the software to indicate nominal vehicle guideway positions for 15 second headway.

WVU - West Virginia University

APPENDIX B - REPORT OF NEW TECHNOLOGY

This report summarizes the existing MPM Phase II redundant computing system design. However, Section 5 suggests some changes which could improve the computing system and increase the system's range of applications.

☆ U. S. GOVERNMENT PRINTING OFFICE: 1980--700-271--209


HE 18.5 .A37 no. DOT- TSC-
UMTA- 80-36
Rucker, Jim T.
Morgantown People Mover
redundant computing system
Form DOT F 1720.2 (8-70)
FORMERLY FORM DOT F 1700.11.1

DOT LIBRARY
00010049